

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
Кафедра АКУСТИКИ ТА АКУСТОЕЛЕКТРОНІКИ

Н. В. Богданова, М. В. Контар

Спеціальні розділи інформатики

Конспект лекцій

Київ 2015

Лекція 1. ЗАГАЛЬНІ ВІДОМОСТІ ПРО ПРОГРАМНИЙ КОМПЛЕКС SIMULINK

Програма SIMULINK є додатком до пакету MATLAB. У певному сенсі SIMULINK можна розглядати як самостійний продукт фірми MathWorks, однак він працює тільки за наявності ядра MATLAB і використовує багато функцій, що входять до його складу.

Слід зазначити, що пакет MATLAB орієнтований в першу чергу на обробку масивів даних (матриць, векторів і т.п.). Це дозволяє істотно підвищити ефективність процедур, що працюють із зазначеними типами даних, в порівнянні з мовами програмування «загального призначення» (Pascal, C і т.п.), і суттєво відрізняє MATLAB від інших систем, таких, як MAPLE, MathCAD, Mathematica. Векторна обробка даних забезпечує високу швидкість обчислень, в більшості випадків позбавляє користувача від написання циклів і гарантує необхідну точність.

Додаток SIMULINK є інструментом, за допомогою якого можна об'єднувати блоки, відповідні окремим елементам динамічної системи в єдине ціле і вивчати їх поведінку в часі.

Розробка моделей засобами SIMULINK (S-моделі) заснована на технології drag-and-drop («перетягнути і залишити»). Для побудови S-моделі використовуються модулі (або блоки), що зберігаються в бібліотеці SIMULINK.

Бібліотека SIMULINK хороша тим, що, з одного боку, забезпечує користувачу доступ до всіх основних можливостей пакету MATLAB, а з іншого - є достатньо самостійною його компонентою, в тому сенсі, що при роботі з нею не обов'язково мати навички у використанні інших інструментів, що входять до складу пакету.

Блоки, що включаються в створювану модель, можуть бути пов'язані один з одним як по інформації, так і по управлінню. Вид зв'язку залежить від типу блоку і логіки роботи моделі. Дані, якими обмінюються блоки, можуть бути скалярними величинами, векторами або матрицями довільної розмірності.

Будь-яка S-модель може мати ієрархічну структуру, тобто складатися з моделей більш низького рівня, причому число рівнів ієрархії практично не обмежена. Поряд з іншими параметрами моделювання користувач може задавати спосіб зміни модельного часу (з постійним або змінним кроком), а також умови закінчення моделювання.

У ході моделювання є можливість спостерігати за процесами, що відбуваються в системі. Для цього використовуються спеціальні «оглядові вікна»,

що входять до складу бібліотеки SIMULINK. Цікавлять користувача характеристики можуть бути представлені як у числовій, так і в графічній формі.

Застосування принципів структурного і модульного програмування дозволяє представляти різні алгоритми у вигляді набору уніфікованих програмних модулів, що покращує видимість програми, полегшує її налагодження і в кінцевому рахунку зменшує загальний обсяг програмного забезпечення, що підлягає розробці. Крім цього складу бібліотеки SIMULINK може бути поповнений користувачем за рахунок розробки власних блоків.

Слід зазначити, що у складі MATLAB є безліч інших додатків, заснованих на методах графічного (візуального) програмування, що допускають спільну роботу з додатком SIMULINK.

В даний час існує велика кількість навчальної літератури з додатком SIMULINK [1 - 5]. Крім цього корисну інформацію можна знайти на сайтах:

<http://matlab.exponenta.ru/simulink/default.php>

<http://www.mathworks.com>

1.1 Запуск Simulink

Для запуску програми необхідно попередньо запустити пакет MATLAB. Основне вікно пакета MATLAB показано на рис. 1.1. Там же показана підказка з'являється у вікні при наведенні покажчика миші на ярлик Simulink в панелі інструментів.

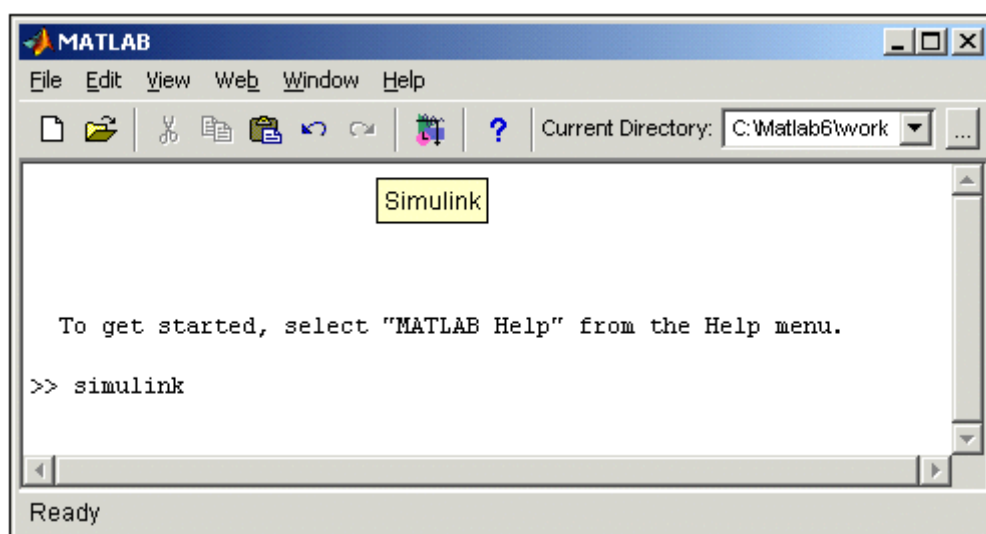



Рис 1.1. Основне вікно програми MATLAB

Після відкриття основного вікна програми MATLAB потрібно запустити програму Simulink. Це можна зробити одним з трьох способів:

- Натиснути кнопку  (Simulink) на панелі інструментів командного вікна MATLAB.
- У командному рядку головного вікна MATLAB надрукувати Simulink і натиснути клавішу Enter на клавіатурі.
- Виконати команду Open в меню File і відкрити файл моделі (mdl - файл).

Останній варіант зручно використовувати для запуску вже готової і налагодженої моделі, коли потрібно лише провести розрахунки і не потрібно додавати нові блоки в модель. Використання першого і другого способів призводить до відкриття вікна оглядача розділів бібліотеки Simulink (рис. 1.2).

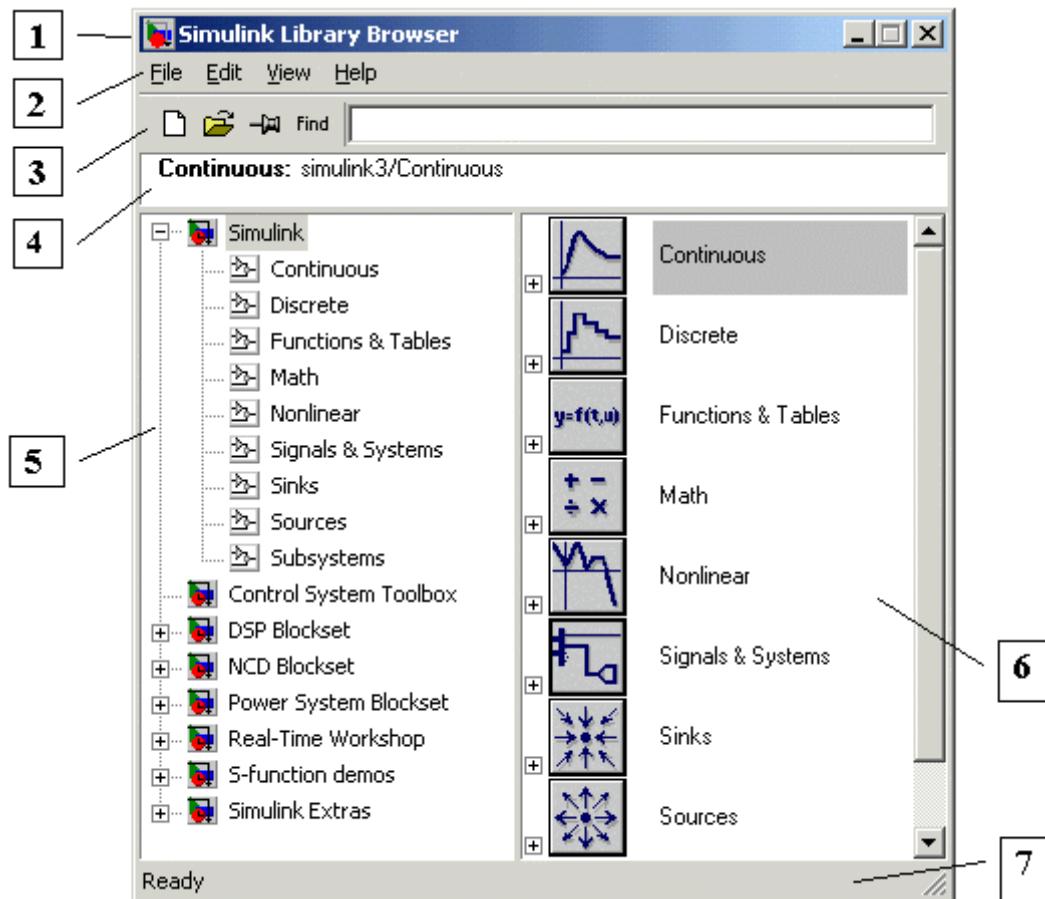


Рис 1.2. Вікно оглядача розділів бібліотеки Simulink

1.2 Огляд розділів бібліотеки Simulink

Вікно оглядача бібліотеки блоків містить такі елементи (Рис. 1.2):

1. Заголовок, з назвою вікна - Simulink Library Browser.
2. Меню, з командами File, Edit, View, Help.
3. Панель інструментів, з ярликами найбільш часто використовуваних команд.
4. Вікно коментаря для виведення пояснюючого повідомлення про обраний блок.
5. Список розділів бібліотеки, реалізований у вигляді дерева.
6. Вікно вмісту розділу бібліотеки (список вкладених розділів бібліотеки або блоків)
7. Рядок стану, що містить підказку по виконуваній дії.

На рис. 2.2 виділена основна бібліотека Simulink (у лівій частині вікна) і показані її розділи (у правій частині вікна).

Бібліотека Simulink містить наступні основні розділи:

1. Continuous - лінійні блоки.
2. Discrete - дискретні блоки.
3. Functions & Tables - функції і таблиці.
4. Math - блоки математичних операцій.
5. Nonlinear - нелінійні блоки.
6. Signals & Systems - сигнали і системи.
7. Sinks - реєструючі пристрої.
8. Sources - джерела сигналів і впливів.
9. Subsystems - блоки підсистем.

Список розділів бібліотеки Simulink представлений у вигляді дерева, і правила роботи з ним є загальними для списків такого виду:

- Піктограма згорнутого вузла дерева містить символ "+", а піктограма розгорнутого містить символ "-".
- Для того щоб розгорнути або згорнути вузол дерева, досить клацнути на його піктограмі лівою клавішею миші (ЛКМ).

При виборі відповідного розділу бібліотеки в правій частині вікна відображається його вміст (рис. 1.3).

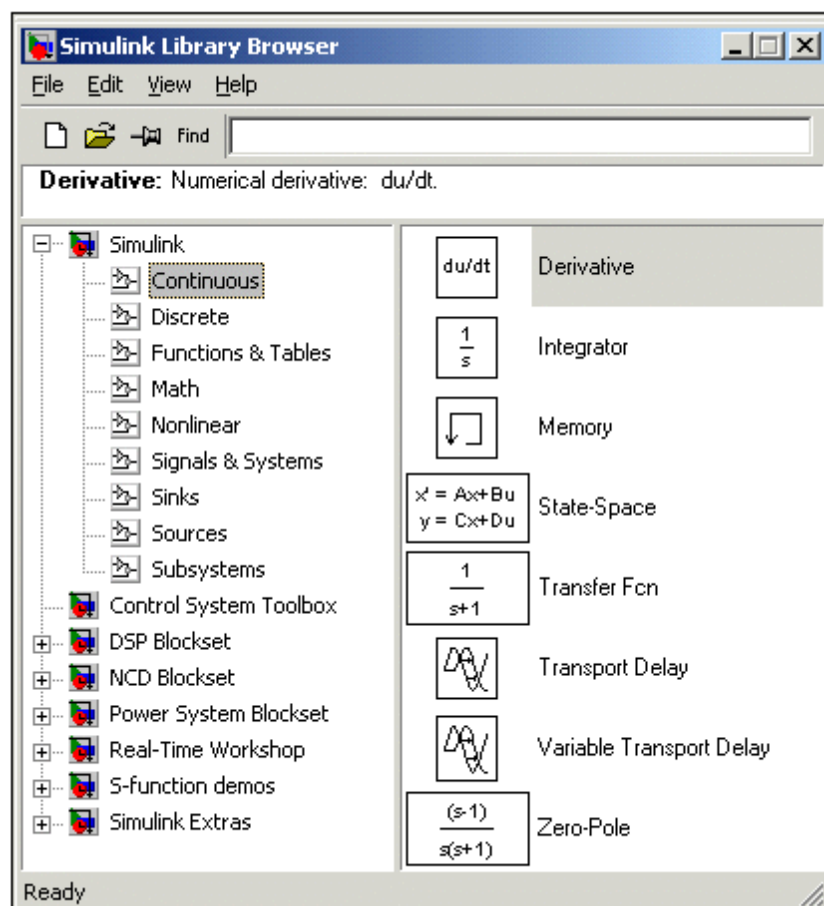


Рис 1.3. Вікно оглядача з набором блоків розділу бібліотеки

Для роботи з вікном використовуються команди зібрані в меню. Меню оглядача бібліотек містить наступні пункти:

- File (Файл) - Робота з файлами бібліотек.
- Edit (Редагування) - Додавання блоків і їх пошук (по назві).
- View (Вид) - Управління показом елементів інтерфейсу.
- Help (Довідка) - Висновок вікна довідки за оглядачеві бібліотек.

Для роботи з оглядачем можна також використовувати кнопки на панелі інструментів (рис. 1.4).

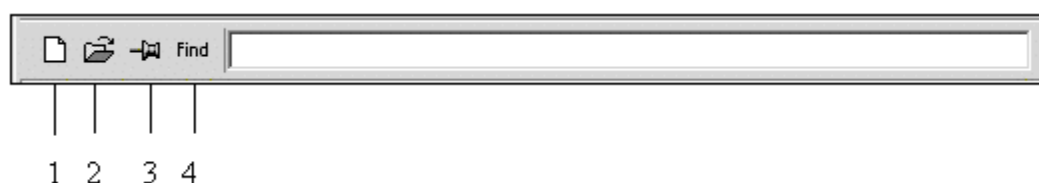


Рис 1.4. Панель інструментів оглядача розділів бібліотек

Кнопки панелі інструментів мають таке призначення:

1. Створити нову S-модель (відкрити нове вікно моделі).
2. Відкрити одну з існуючих S-моделей.
3. Змінити властивості вікна оглядача. Дана кнопка дозволяє встановити режим відображення вікна оглядача "поверх всіх вікон". Повторне натискання скасовує такий режим.
4. Пошук блоку за назвою (по перших символах назви). Після того як блок буде знайдений, у вікні оглядача відкриється відповідний розділ бібліотеки, а блок буде виділено. Якщо ж блок з такою назвою відсутня, то у вікні коментаря буде виведено повідомлення Not found <ім'я блоку> (Блок не знайдено).

Розглянемо розповсюджені блоки SIMULINK.

Для будь-якого блоку за подвійним клацанням миші відкривається вікно параметрів Function Block Parameters, де можна змінювати установки.

Джерела сигналів (Sources)

На рис. 1.5 представлені деякі джерела сигналів.

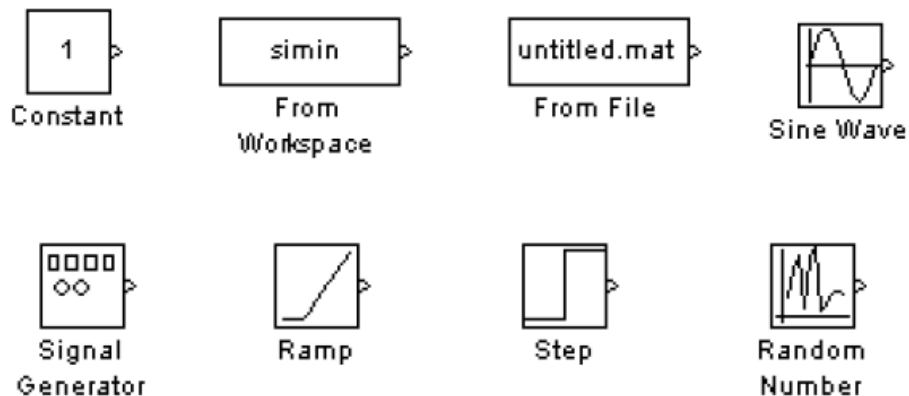


Рис. 1.5. Джерела сигналів

Constant задає постійний за рівнем сигнал;

From Workspace - блок зчитування даних з робочої області;

From File - блок зчитування даних з файлу;

Sine Wave формує синусоїдальний сигнал із заданою частотою, амплітудою, фазою і зміщенням;

Signal Generator формує один з чотирьох видів періодичних сигналів (синусоїдальний, прямокутний, пилкоподібний, випадковий сигнали); 9

Ramp формує лінійний сигнал;

Step формує ступінчастий сигнал;

Random Number формує випадковий сигнал з нормальним розподілом рівня сигналу.

Приймачі сигналів (Sinks)

На рис. 1.6 представлені блоки-приймачі сигналів

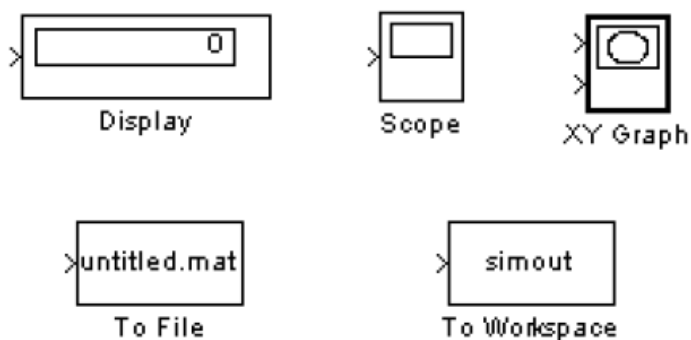


Рис. 1.6. Приймачі сигналів

Цифровий дисплей (**Display**) відображає значення сигналу у вигляді числа;

Віртуальний осцилограф (**Scope**) будує графік сигналу від функції часу;

Віртуальний графобудівник (**XY Graph**) будує графік одного сигналу в залежності від іншого (графік виду $Y(X)$);

Блок «**To File**» записує дані, що надходять на його вхід, в файл;

Блок «**To Workspace**» записує дані, що надходять на його вхід, в робочу область MATLAB.

Блоки математичних операцій

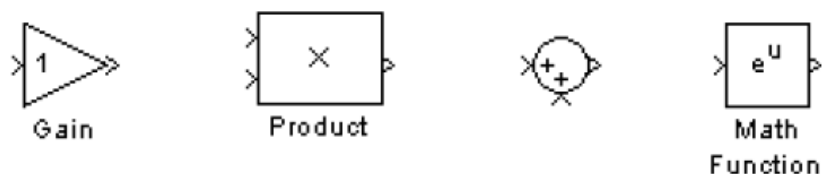


Рис. 1.3. Блоки математичних операцій

Блок **Gain** примножує вхідний сигнал на заданий коефіцієнт.

Блок **Product** перемножує два (або більше) сигналів.

Блок **Sum** підсумовує два (або більше) сигналів.

Блок **Math Function** перетворює вхідний сигнал по заданій зі списку функції (наприклад, зведення в квадрат, корінь квадратний, логарифм і т.д.).

Відзначимо ще деякі часто зустрічаються блоки (рис. 1.7).

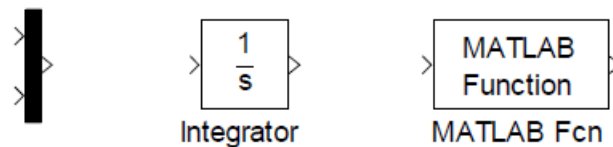


Рис. 1.7. Приклади часто зустрічаючихся функцій


Блок **Mux** (мультиплексор) об'єднує сигнали. Піктограма мультиплексора показані першими зліва на рис. 1.7.

Блок **Integrator** інтегрує сигнал.

Блок **MATLAB Fcn** формує функцію, задану користувачем на мові MatLab.

1.3 Створення моделі

Для створення моделі в середовищі SIMULINK необхідно послідовно виконати ряд дій:

1. Створити новий файл моделі за допомогою команди File / New / Model, або використовуючи кнопку  на панелі інструментів (тут і далі, за допомогою символу "/", вказані пункти меню програми, які необхідно послідовно вибрати для виконання зазначеної дії). Новостворене вікно моделі показано на рис. 1.8.

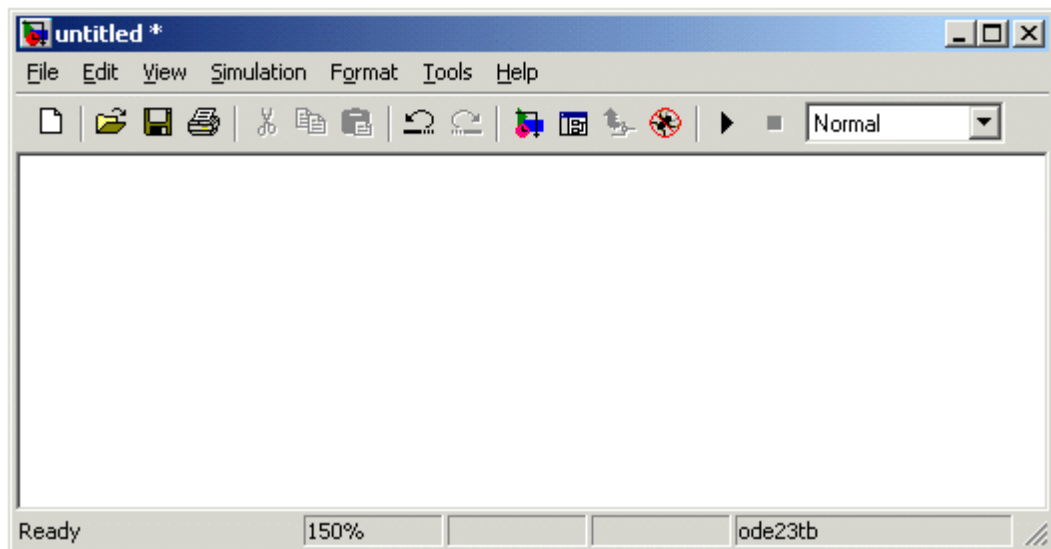


Рис 1.8. Порожнє вікно моделі

2. Розташувати блоки у вікні моделі. Для цього необхідно відкрити відповідний розділ бібліотеки (Наприклад, Sources - Джерела). Далі, вказавши курсором на необхідний блок і натиснувши на ліву клавішу "миші" - "перетягнути" блок у створене вікно. Клавішу миші потрібно тримати натиснутою. На рис 1.9 показано вікно моделі, що містить блоки.

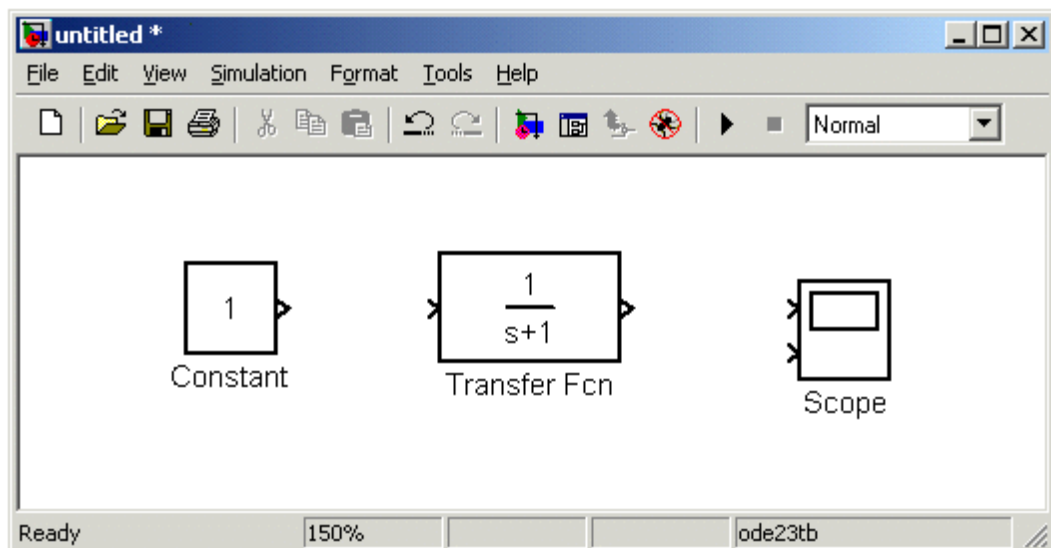


Рис 1.9. Вікно моделі, що містить блоки

Для видалення блоку необхідно вибрати блок (вказати курсором на його зображення і натиснути ліву клавішу "миші"), а потім натиснути клавішу Delete на клавіатурі.

Для зміни розмірів блоку потрібно вибрати блок, встановити курсор в один з кутів блоку і, натиснувши ліву клавішу "миші", змінити розмір блоку (курсор при цьому перетвориться в двосторонню стрілку).

3. Далі, якщо це потрібно, потрібно змінити параметри блоку, встановлені програмою "за замовчуванням". Для цього необхідно двічі клацнути лівою клавішею "миші", вказавши курсором на зображення блоку. Відкриється вікно редагування параметрів даного блоку. При завданні чисельних параметрів слід мати на увазі, що в якості десяткового роздільника повинна використовуватися точка, а не кома. Після внесення змін потрібно закрити вікно кнопкою OK. На рис.1.10 як приклад показані блок, що моделює передавальну функцію і вікно редагування параметрів даного блоку.

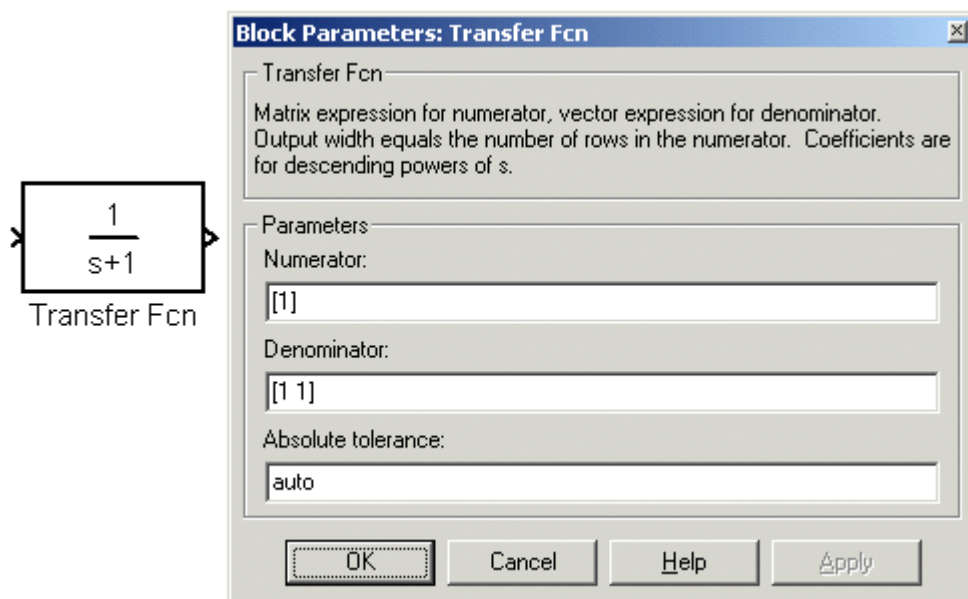


Рис 1.10. Блок, що моделює передавальну функцію і вікно редагування параметрів блоку

4. Після установки на схемі всіх блоків із потрібних бібліотек потрібно виконати з'єднання елементів схеми. Для з'єднання блоків необхідно вказати курсором на "вихід" блоку, а потім, натиснувши і, не відпускаючи ліву клавішу "миші", провести лінію до входу іншого блоку. Після чого відпустити клавішу. У разі правильного з'єднання зображення стрілки на вході блоку змінює колір. Для створення точки

розгалуження в з'єднувальній лінії потрібно підвести курсор до передбачуваного вузлу і, натиснувши праву кнопку "миші", протягнути лінію. Для видалення лінії потрібно вибрати лінію (так само, як це виконується для блоку), а потім натиснути кнопку Delete на клавіатурі. Схема моделі, в якій виконані з'єднання між блоками, показана на рис. 1.11.

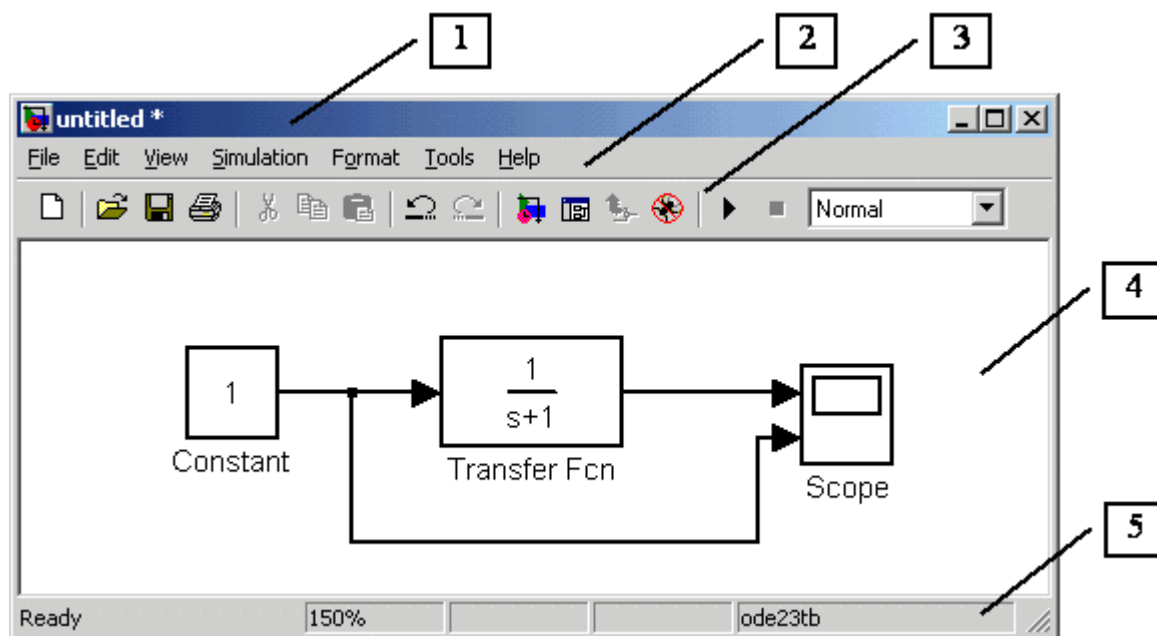


Рис 1.11. Схема моделі

5. Після складання розрахункової схеми необхідно зберегти її у вигляді файлу на диску, вибравши пункт меню File / Save As ... у вікні схеми і вказавши папку та ім'я файлу. Слід мати на увазі, що ім'я файлу не повинен перевищувати 32 символів, повинно починатися з літери і не може містити символи кирилиці і спецсимволи. Ця ж вимога стосується і шляхи файлу (до тих папок, в яких зберігається файл). При подальшому редагуванні схеми можна користуватися пунктом меню File / Save. При повторних запусках програми SIMULINK завантаження схеми здійснюється за допомогою меню File / Open ... у вікні оглядача бібліотеки або з основного вікна MATLAB.

1.4 Вікно моделі

Вікно моделі містить такі елементи (див. рис. 1.11):

1. Заголовок, з назвою вікна. Новоствореному вікну присвоюється ім'я Untitled з відповідним номером.

2. Меню з командами File, Edit, View і т.д.
3. Панель інструментів.
4. Вікно для створення схеми моделі.
5. Рядок стану, що містить інформацію про поточний стан моделі.

Меню вікна містить команди для редагування моделі, її налаштування і управління процесом розрахунку, роботи файлами і т. п.:

- File (Файл) - Робота з файлами моделей.
- Edit (Редагування) - Зміна моделі і пошук блоків.
- View (Вид) - Управління показом елементів інтерфейсу.
- Simulation (Моделювання) - Завдання налаштувань для моделювання та керування процесом розрахунку.
- Format (Форматування) - Зміна зовнішнього вигляду блоків і моделі в цілому.
- Tools (Інструментальні засоби) - Застосування спеціальних засобів для роботи з моделлю (відладчик, лінійний аналіз і т.п.)
- Help (Довідка) - Висновок вікон довідкової системи.

Для роботи з моделлю можна також використовувати кнопки на панелі інструментів (рис. 1.12).

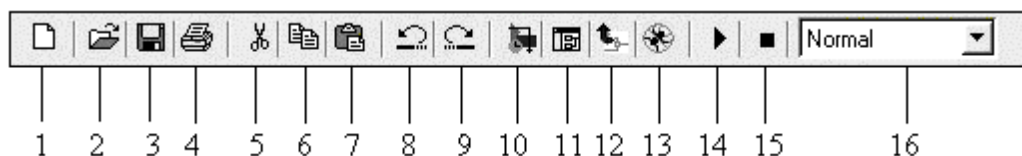


Рис 1.12. Панель інструментів вікна моделі

Кнопки панелі інструментів мають таке призначення:

1. New Model - Відкрити нове (пусте) вікно моделі.
2. Open Model - Відкрити існуючий mdl-файл.
3. Save Model - Зберегти mdl-файл на диску.
4. Print Model - Вивід на друк блок-діаграми моделі.
5. Cut - Вирізати виділену частину моделі в буфер проміжного зберігання.
6. Copy - Скопіювати виділену частину моделі в буфер проміжного зберігання.
7. Paste - Вставити у вікно моделі вміст буфера проміжного зберігання.
8. Undo - Скасувати попередню операцію редагування.
9. Redo - Відновити результат скасованої операції редагування.
10. Library Browser - Відкрити вікно оглядача бібліотек.

11. Toggle Model Browser - Відкрити вікно оглядача моделі.
 12. Go to parent system - Перехід з підсистеми в систему вищого рівня ієрархії ("родительську систему"). Команда доступна тільки, якщо відкрита підсистема.
 13. Debug - Запуск відладчика моделі.
 14. Start / Pause / Continue Simulation - Запуск моделі на виконання (команда Start); після запуску моделі на зображенні кнопки виводиться символ, і їй відповідає вже команда Pause (Призупинити моделювання); для відновлення моделювання слід клацнути по тій же кнопці, оскільки в режимі паузи їй відповідає команда Continue (Продовжити).
 15. Stop - Закінчити моделювання. Кнопка стає доступною після початку моделювання, а також після виконання команди Pause.
 16. Normal / Accelerator - Звичайний / Прискорений режим розрахунку. Інструмент доступний, якщо встановлено додаток Simulink Performance Tool.
- У нижній частині вікна моделі знаходиться рядок стану, в якій відображаються короткі коментарі до кнопок панелі інструментів, а також до пунктів меню, коли покажчик миші знаходиться над відповідним елементом інтерфейсу. Це ж текстове поле використовується і для індикації стану Simulink: Ready (Готовий) або Running (Виконання). У рядку стану відображаються також:
- масштаб відображення блок-діаграми (у відсотках, вихідне значення дорівнює 100%),
 - індикатор ступеня завершеності сеансу моделювання (з'являється після запуску моделі),
 - поточне значення модельного часу (виводиться також тільки після запуску моделі),
 - використовуваний алгоритм розрахунку станів моделі (метод рішення).

1.5 Основні прийоми підготовки і редагування моделі

1.5.1. Додавання текстових написів

Для підвищення наочності моделі зручно використовувати текстові написи. Для створення напису потрібно вказати мишею місце написи і двічі клацнути лівою клавішею миші. Після цього з'явиться прямокутна рамка з курсором вводу. Аналогічним чином можна змінити і підписи до блоками моделей. На рис. 1.13 показані текстова напис і зміна напису в блоці передавальної функції. Слід мати на увазі, що розглянута версія програми (Simulink 4) не адаптована до використання кирилических шрифтів, і застосування їх може мати самі різні наслідки: - відображення

написів в нечитабельним вигляді, обрізання написів, повідомлення про помилки, а також неможливість відкрити модель після її збереження. Тому, застосування написів російською мовою для поточної версії Simulink вкрай не бажано.

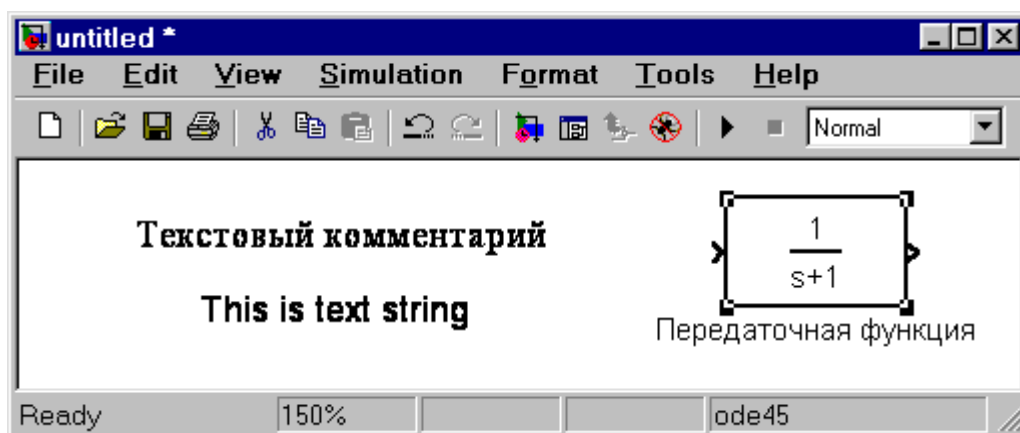


Рис. 1.13. Текстова напис і зміна напису в Transfer Function

Відображення малюнків у вікні моделі

Для підвищення наочності моделі може знадобитися помістити у вікні моделі малюнок

Для цього необхідно:

1. Створити порожню підсистему (блок Subsystem розділу Ports&Subsystems бібліотеки Simulink).
2. Виконати її маскування (Edit\Mask Subsystem).
3. Відкрити вікно редагування маски (Edit \ Edit mask ...).
4. На вкладці Icon Drawing Commands вказати команду для зчитування з файлу і відображення графічного образу, наприклад:

```
image (imread ('Picture_1.bmp', 'bmp'))
```

де перший параметр - ім'я графічного файлу, а другий - його тип.

На рис. 1.14 показаний приклад вікна моделі із створеними таким чином малюнками.

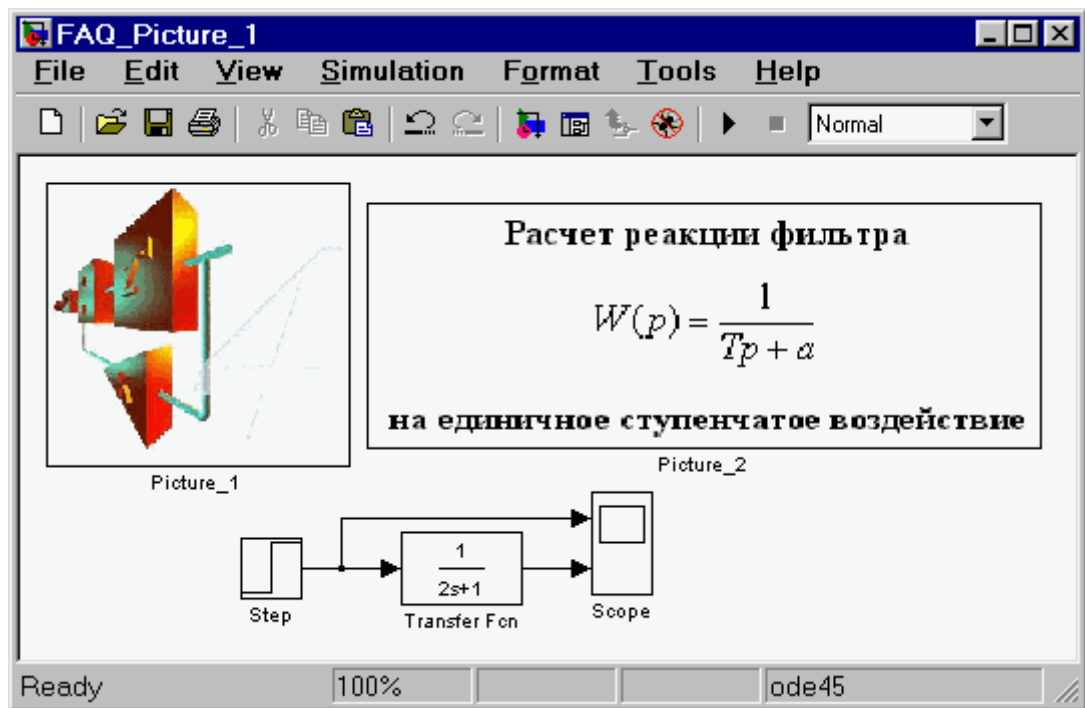


Рис 1.14. Приклад моделі з малюнками

1.5.2. Виділення об'єктів

Для виконання якої-небудь дії з елементом моделі (блоком, сполучної лінією, написом) цей елемент необхідно спочатку виділити.

Виділення об'єктів найпростіше здійснюється мишею. Для цього необхідно встановити курсор миші на потрібному об'єкті і клацнути лівою клавшею миші. Відбудеться виділення об'єкта. Про це будуть свідчити маркери по кутах об'єкта (див. рис. 1.13). Можна також виділити кілька об'єктів. Для цього треба встановити курсор миші поблизу групи об'єктів, натиснути ліву кнопку миші і, не відпускаючи її, почати переміщати мишу. З'явиться пунктирна рамка, розміри якої будуть змінюватися при переміщенні миші. Усі охоплені рамкою об'єкти стають виділеними. Виділити всі об'єкти також можна, використовуючи команду Edit / Select All. Після виділення об'єкта його можна копіювати або переміщати в буфер проміжного зберігання, витягати з буфера, а також видаляти, використовуючи стандартні прийоми роботи в Windows-програмах.

1.5.3. Копіювання і переміщення об'єктів в буфер проміжного зберігання

Для копіювання об'єкта в буфер його необхідно попередньо виділити, а потім виконати команду Edit / Copy або скористатися інструментом на панелі інструментів. Для вирізання об'єкта в буфер його необхідно попередньо виділити, а потім виконати команду Edit / Cut або скористатися інструментом на панелі інструментів.

При виконанні даних операцій слід мати на увазі, що об'єкти поміщаються у власний буфер MATLAB і недоступні з інших додатків. Використання команди `Edit / Copy model to Clipboard` дозволяє помістити графічне зображення моделі в буфер Windows і, відповідно, робить його доступним для решти програм.

Копіювання можна виконати і таким чином: натиснути праву клавішу миші, і не відпускаючи її, перемістити об'єкт. При цьому буде створена копія об'єкта, яку можна перемістити в потрібне місце.

1.5.4. Вставка об'єктів з буфера проміжного зберігання

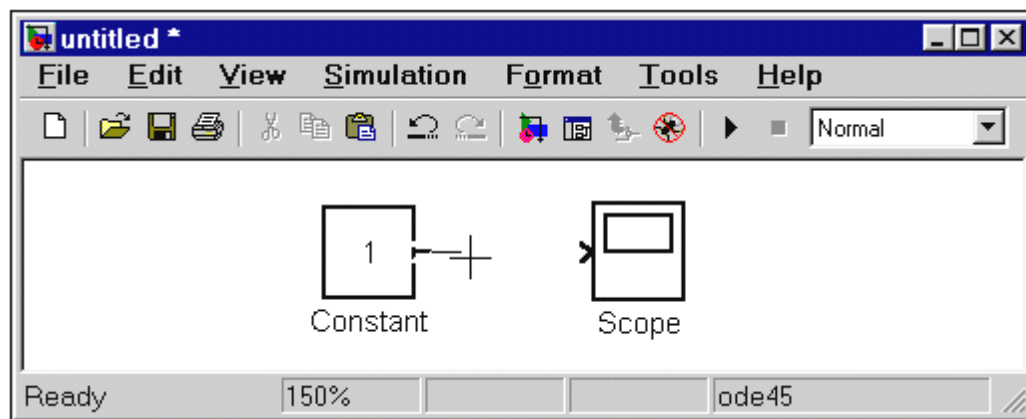
Для вставки об'єкта з буфера необхідно попередньо вказати місце вставки, клацнувши лівою клавішею миші в передбачуваному місці вставки, а потім виконати команду `Edit / Paste` або скористатися інструментом на панелі інструментів.

1.5.5. Видалення об'єктів

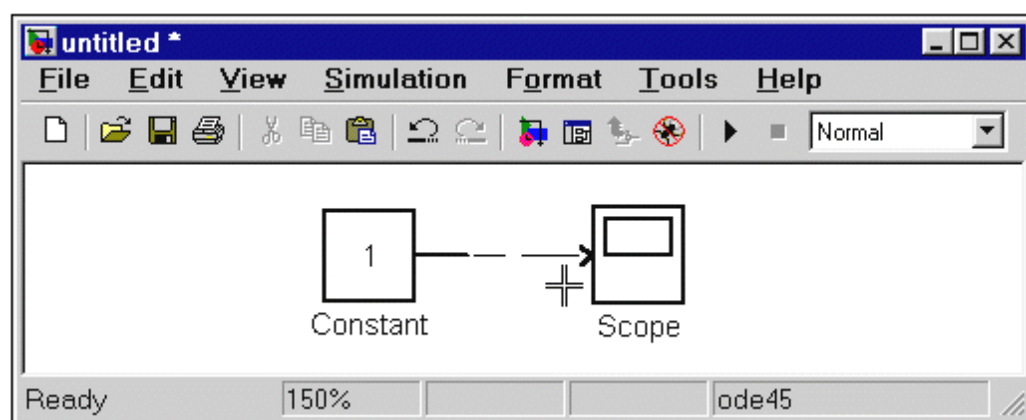
Для видалення об'єкта його необхідно попередньо виділити, а потім виконати команду `Edit / Clear` або скористатися клавішею `Delete` на клавіатурі. Слід врахувати, що команда `Clear` видаляє блок без приміщення його в буфер обміну. Однак цю операцію можна скасувати командою меню `File / Undo`.

1.5.6. З'єднання блоків

Для з'єднання блоків необхідно спочатку встановити курсор миші на вихідний порт одного з блоків. Курсор при цьому перетвориться на великий хрест з тонких ліній (рис. 1.15, а). Тримавши ліву кнопку миші, потрібно перемістити курсор до вхідного порту потрібного блоку. Курсор миші прийме вигляд хреста з тонких здвоєних ліній (рис. 1.15, б). Після створення лінії необхідно відпустити ліву кнопку миші. Свідченням того, що з'єднання створено, буде жирна стрілка біля вхідного порту блоку. Виділення лінії відбувається точно так як і виділення блоку - одинарним клацанням лівої клавіші миші. Чорні маркери, розташовані у вузлах сполучної лінії будуть говорити про те, що лінія виділена.



а



б

Рис. 1.15. Початок та завершення створення з'єднання

Створення петлі лінії з'єднання виконується також як переміщення блоку. Лінія з'єднання виділяється, і потім потрібна частина лінії переміщується. Малюнок 1.16 пояснює цей процес.

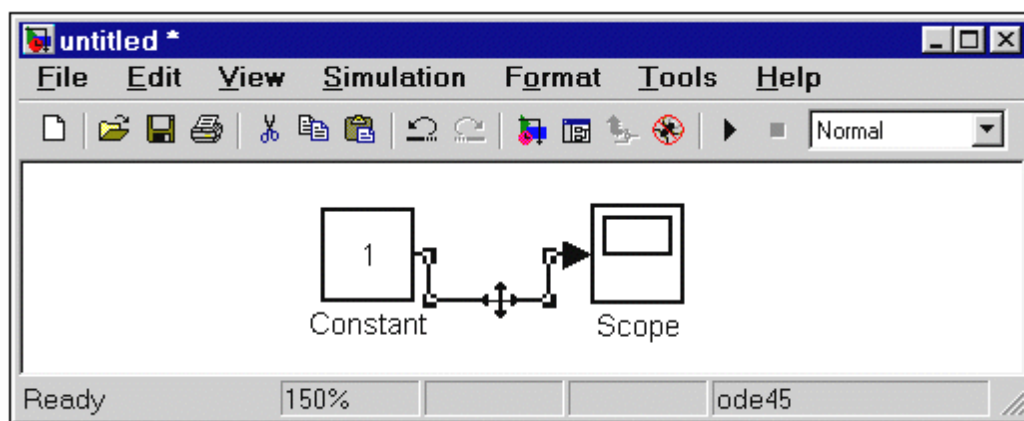


Рис 1.16. Створення петлі в з'єднувальній лінії

Видалення з'єднань виконується так як і будь-яких інших об'єктів.

1.5.7. Зміна розмірів блоків

Для зміни розміру блоку він виділяється, після чого курсор миші треба встановити на один із маркерів по кутах блоку. Після перетворення курсора в двосторонню стрілку, необхідно натиснути ліву кнопку миші і розтягнути (або стиснути) зображення блоку. На рис. 1.17 показаний цей процес. Розміри написів блоку при цьому не змінюються.

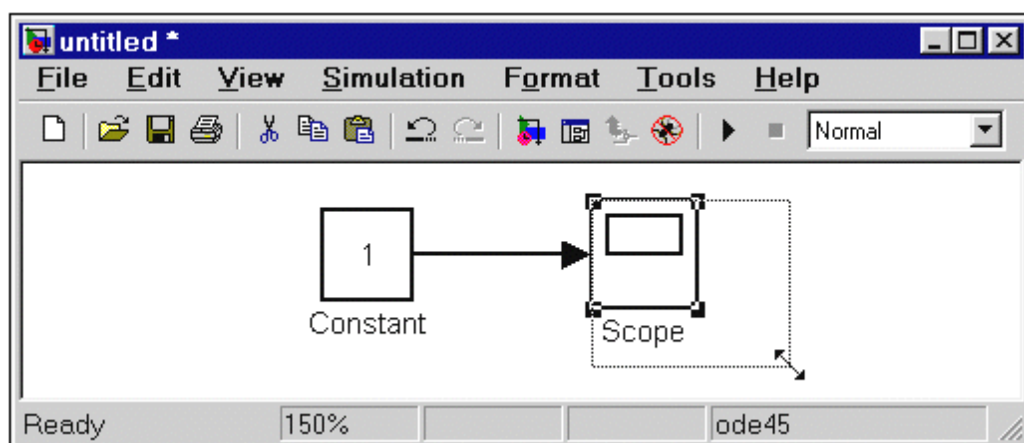


Рис. 1.17. Зміна розміру блоку



1.5.8. Переміщення блоків

Будь блок моделі можна перемістити, виділивши його, і пересунувши, тримаючи натиснутою ліву кнопку миші. Якщо до входів і виходів блоку підведені сполучні лінії, то вони не розриваються, а лише скорочуються або збільшуються в довжині. У

з'єднання можна також вставити блок, що має один вхід і один вихід. Для цього його потрібно розташувати в необхідному місці сполучної лінії.

1.5.9. Використання команд Undo і Redo

У процесі освоєння програми користувач може вчиняти дії здаються йому незворотними (наприклад, випадкове видалення частини моделі, копіювання і т.д.). У цьому випадку слід скористатися командою Undo - скасування останньої операції.

Команду можна викликати за допомогою кнопки  в панелі інструментів вікна моделі або з меню Edit. Для відновлення скасованої операції служить команда Redo (інструмент .

1.5.10. Форматування об'єктів

У меню Format (також як і в контекстному меню, що викликається натисканням правої клавіші миші на об'єкті) знаходиться набір команд форматування блоків. Команди форматування поділяються на кілька груп:

1. Зміна відображення написів:

- Font - Форматування шрифту написів і текстових блоків.
- Text alignment - Вирівнювання тексту в текстових написах.
- Flip name - Переміщення підпису блоку.
- Show / Hide name - Відображення або приховування підпису блоку.

2.Змінення кольорів відображення блоків:

- Foreground color - Вибір кольору ліній для виділених блоків.
- Background color - Вибір кольору фону виділених блоків.
- Screen color - Вибір кольору фону для всього вікна моделі.

3. Зміна положення блоку і його виду:

- Flip block - Дзеркальне відображення щодо вертикальної осі симетрії.
- Rotate block - Поворот блоку на 900 за годинниковою стрілкою.
- Show drop shadow - Показ тіні від блоку.
- Show port labels - Показ міток портів.

4. Інші установки:

- Library link display - Показ зв'язків з бібліотеками.
- Sample time colors - Вибір кольору блоку індикації часу.
- Wide nonscalar lines - Збільшення / зменшення ширини не скалярних ліній.
- Signal dimensions - Показ розмірності сигналів.
- Port data types - Показ даних про тип портів.
- Storage class - Клас пам'яті. Параметр, встановлюваний при роботі Real-Time Workshop.

- Execution order - Висновок порядкового номера блоку в послідовності виконання.

1.6 Встановлення параметрів розрахунку і його виконання

Перед виконанням розрахунків необхідно попередньо задати параметри розрахунку. Завдання параметрів розрахунку виконується в панелі управління меню Simulation / Parameters. Вид панелі управління наведено на рис. 1.18.

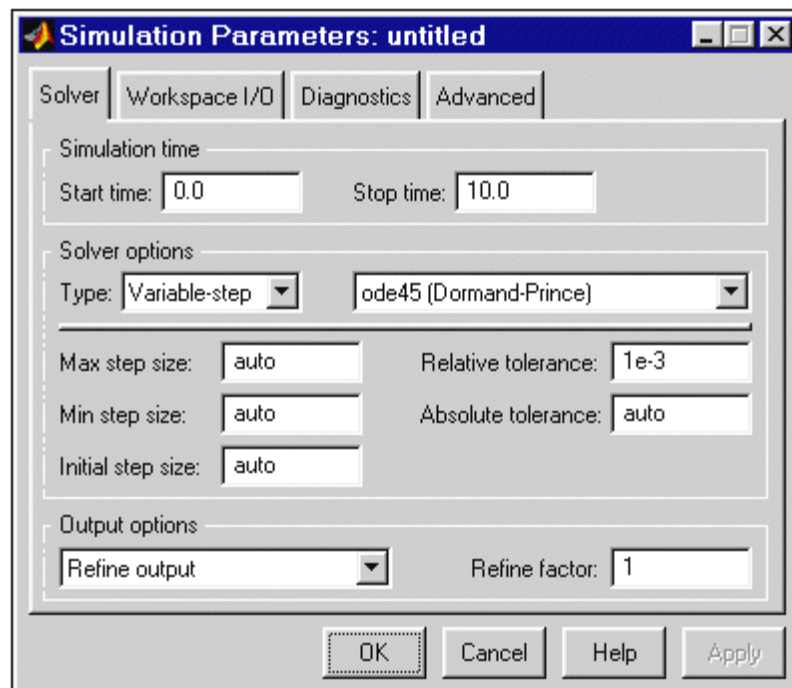


Рис 1.18. Панель управління

Вікно налаштування параметрів розрахунку має 4 вкладки:

- Solver (Розрахунок) - Установка параметрів розрахунку моделі.
- Workspace I / O (Введення / виведення даних в робочу область) - Установка параметрів обміну даними з робочою областю MATLAB.
- Diagnostics (Діагностика) - Вибір параметрів діагностичного режиму.
- Advanced (Додатково) - Встановлення додаткових параметрів.

Установка параметрів розрахунку моделі виконується за допомогою елементів управління, розміщених на вкладці Solver. Ці елементи розділені на три групи (рис. 1.18): Simulation time (Інтервал моделювання або, іншими словами, час розрахунку), Solver options (Параметри розрахунку), Output options (Параметри виводу).

1.6.1. Установка параметрів розрахунку моделі

1.6.1.1. Simulation time (Інтервал моделювання або час розрахунку)

Час розрахунку задається вказівкою початкового (Start time) і кінцевого (Stop time) значень часу розрахунку. Початковий час, як правило, задається рівним нулю. Величина кінцевого часу задається користувачем виходячи з умов розв'язуваної задачі.

1.6.1.2. Solver options (Параметри розрахунку)

При виборі параметрів розрахунку необхідно вказати спосіб моделювання (Type) і метод розрахунку нового стану системи. Для параметра Type доступні два варіанти - з фіксованим (Fixed-step) або з перемінним (Variable-step) кроком. Як правило, Variable-step використовується для моделювання безперервних систем, а Fixed-step - для дискретних.

Список методів розрахунку нового стану системи містить кілька варіантів. Перший варіант (discrete) використовується для розрахунку дискретних систем. Решта методи використовуються для розрахунку безперервних систем. Ці методи різні для змінного (Variable-step) і для фіксованого (Fixed-step) кроку часу, але, по суті, являють собою процедури розв'язання систем диференціальних рівнянь. Детальний опис кожного з методів розрахунку станів системи приведено у вбудованій довідковій системі MATLAB.

Нижче двох списків, що розкриваються Type знаходиться область, вміст якої змінюється залежно від вибраного способу зміни модельного часу. При виборі Fixed-step в даній області з'являється текстове поле Fixed-step size (величина фіксованого кроку) дозволяє вказувати величину кроку моделювання (див. рис. 1.19). Величина кроку моделювання за умовчанням встановлюється системою автоматично (auto). Необхідна величина кроку може бути введена замість значення auto або у формі числа, або у вигляді обчислюваного виразу (те ж саме відноситься і до всіх параметрів встановлюються системою автоматично).

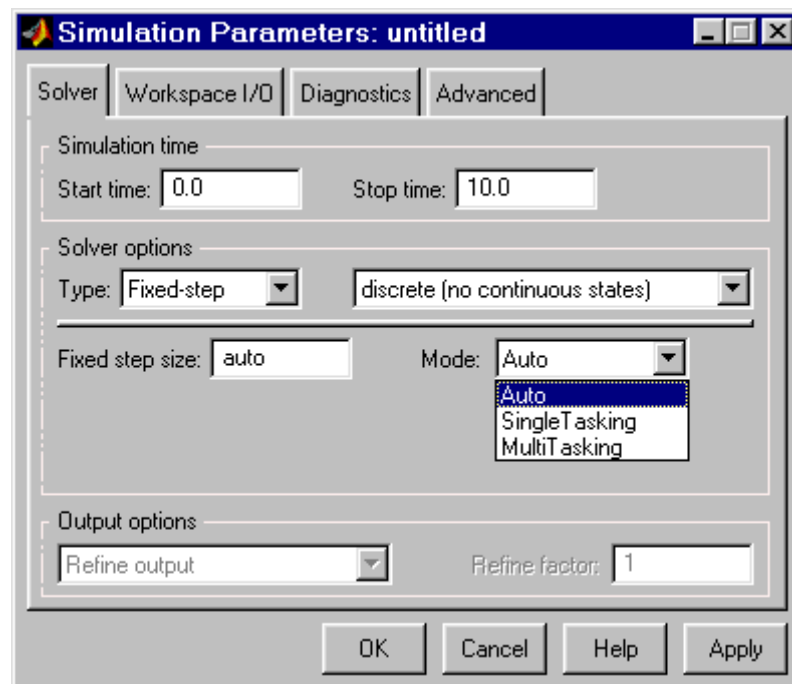


Рис 1.19. Вкладка Solver при виборі фіксованого кроку розрахунку

При виборі Fixed-step необхідно також задати режим розрахунку (Mode). Для параметра Mode доступні три варіанти:

- MultiTasking (Багатозадачний) - необхідно використовувати, якщо в моделі присутні паралельно працюючі підсистеми, і результат роботи моделі залежить від часових параметрів цих підсистем. Режим дозволяє виявити невідповідність швидкості і дискретності сигналів, що пересилаються блоками один одному.
- SingleTasking (однозадачний) - використовується для тих моделей, в яких недостатньо суворя синхронізація роботи окремих складових не впливає на кінцевий результат моделювання.
- Auto (Автоматичний вибір режиму) - дозволяє Simulink автоматично встановлювати режим MultiTasking для тих моделей, в яких використовуються блоки з різними швидкостями передачі сигналів і режим SingleTasking для моделей, в яких містяться блоки, які оперують однаковими швидкостями.

При виборі Variable-step в області з'являються поля для установки трьох параметрів:

- Max step size - максимальний крок розрахунку. За замовчуванням він встановлюється автоматично (auto) і його значення в цьому випадку дорівнює $(SfopTime - StartTime)/50$. Довольно часто це значення виявляється занадто

великим, і спостережувані графіки являють собою ламані (а не плавні) лінії. У цьому випадку величину максимального кроку розрахунку необхідно задавати явно.

- Min step size - мінімальний крок розрахунку.
- Initial step size - початкове значення кроку моделювання.

При моделюванні безперервних систем з використанням змінного кроку необхідно вказати точність обчислень: відносну (Relative tolerance) і абсолютну (Absolute tolerance). За умовчанням вони дорівнюють відповідно 10^3 і auto.

1.6.1.3. Output options (Параметри виводу)

У нижній частині вкладки Solver задаються налаштування параметрів виведення вихідних сигналів модельованої системи (Output options). Для даного параметра можливий вибір одного з трьох варіантів:

- Refine output (Скоригований вивід) - дозволяє змінювати дискретність реєстрації модельного часу і тих сигналів, які зберігаються в робочій області MATLAB за допомогою блоку To Workspace. Установка величини дискретності виконується в рядку редагування Refine factor, розташованої праворуч. За замовчуванням значення Refine factor дорівнює 1, це означає, що реєстрація проводиться за кроком рівним 1 (тобто для кожного значення модельного часу). Якщо задати Refine factor дорівнює 2, це означає, що буде реєструватися кожне друге значення сигналів, 3 - кожне третє т. д. Параметр Refine factor може приймати тільки цілі позитивні значення
- Produce additional output (Додатковий вивід) - забезпечує додаткову реєстрацію параметрів моделі в задані моменти часу; їх значення вводяться в рядку редагування (в цьому випадку вона називається Output times) у вигляді списку, укладеного в квадратні дужки. При використанні цього варіанта базовий крок реєстрації дорівнює 1. Значення часу в списку Output times можуть бути дробовими числами і мати будь-яку точність.
- Produce specified output only (Формувати тільки заданий вивід) - встановлює вивід параметрів моделі тільки в задані моменти часу, які вказуються в полі Output times (Моменти часу виводу).

1.6.2. Установка параметрів обміну з робочою областю

Елементи, що дозволяють управляти введенням і виведенням в робочу область MATLAB проміжних даних і результатів моделювання, розташовані на вкладці Workspace I / O (рис. 1.20).

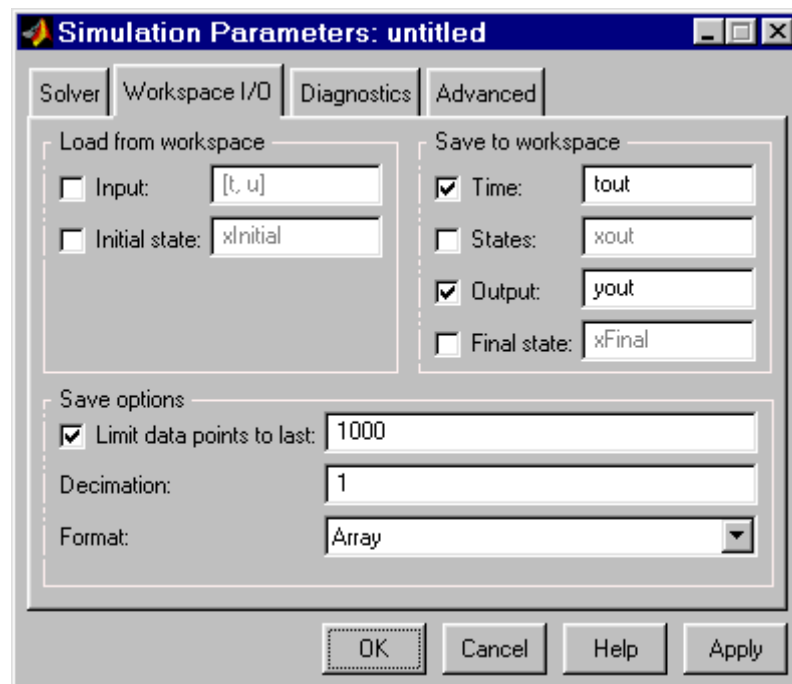


Рис 1.20. Вкладка Workspace I / O діалогового вікна установки параметрів моделювання

Елементи вкладки розділені на 3 поля:

- Load from workspace (Завантажити з робочою областю). Якщо прапорець Input (Вхідні дані) встановлено, то в розташованому праворуч текстовому полі можна ввести формат даних, які будуть зчитуватися з робочої області MATLAB. Установка прапорця Initial State (Початковий стан) дозволяє ввести в пов'язаному з ним текстовому полі ім'я змінної, що містить параметри початкового стану моделі. Дані, зазначені в полях Input і Initial State, передаються у виконувану модель за допомогою одного або більше блоків In (з розділу бібліотеки Sources).
- Save to workspace (Записати в робочу область) - Дозволяє встановити режим виводу значень сигналів в робочу область MATLAB і задати їх імена.
- Save options (Параметри запису) - Задає кількість рядків при передачі змінних в робочу область. Якщо прапорець Limit rows to last встановлений, то в полі введення можна вказати кількість переданих рядків (відлік рядків виробляється від моменту завершення розрахунку). Якщо прапорець не встановлений, то передаються всі дані. Параметр Decimation задає крок запису змінних в робочу область (аналогічно параметру Refine factor вкладки Solver). Параметр Format (формат даних) задає формат переданих в робочу область даних. Доступні формати Array (Масив), Structure (Структура), Structure With Time (Структура з додатковим полем - "час").

1.6.3. Установка параметрів діагностування моделі

Вкладка Diagnostics (рис. 1.21) дозволяє змінювати перелік діагностичних повідомлень, що виводяться Simulink в командному вікні MATLAB, а також встановлювати додаткові параметри діагностики моделі.

Повідомлення про помилки або проблемних ситуаціях, виявлених Simulink в ході моделювання і вимагають втручання розробника виводяться в командному вікні MATLAB. Початковий перелік таких ситуацій і вид реакції на них наведено у списку на вкладці Diagnostics. Розробник може вказати вид реакції на кожне з них, використовуючи групу перемикачів у полі Action (вони стають доступні, якщо в списку обрано одне з подій):

- None - ігнорувати,
- Warning - видати попередження і продовжити моделювання,
- Error - видати повідомлення про помилку і зупинити сеанс моделювання.

Обраний вид реакції відображається в списку поруч із найменуванням події.

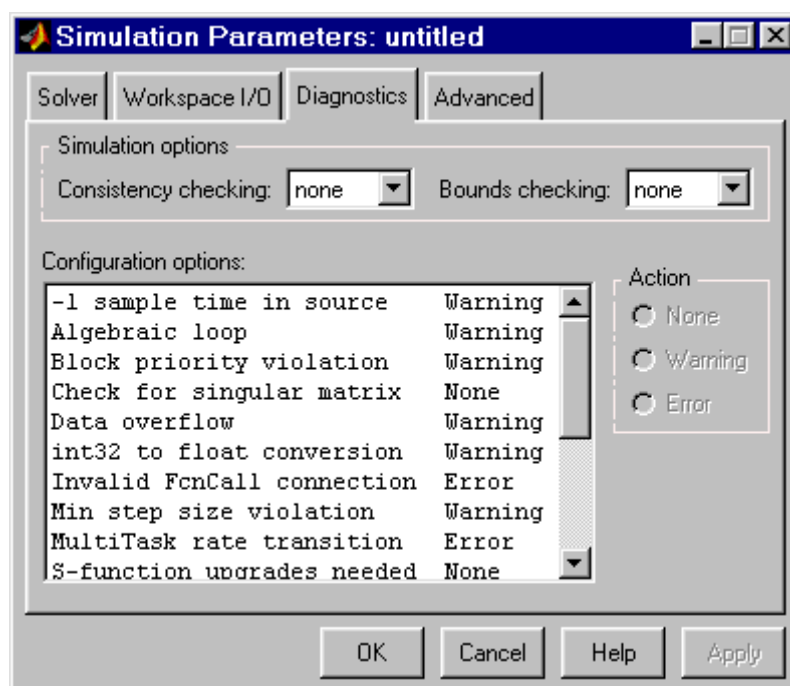




Рис 1.21. Вкладка Diagnostics вікна установки параметрів моделювання

1.6.4. Виконання розрахунку

Запуск розрахунку виконується за допомогою вибору пункту меню Simulation / Start або  інструменту на панелі інструментів. Процес розрахунку можна завершити достроково, вибравши пункт меню Simulation / Stop або інструменту  на панелі

інструментів. Розрахунок також можна зупинити (Simulation / Pause) і потім продовжити (Simulation / Continue).

1.7 Завершення роботи

Для завершення роботи необхідно зберегти модель у файлі, закрити вікно моделі, вікно оглядача бібліотек, а також основне вікно пакета MATLAB.

1.8. Побудова найпростіших моделей

Розглянемо приклад, на якому покажемо, як будується модель сигналу виду $x(t) = 0,5 * \sin \pi t + t^2$ на інтервалі, і відобразимо його на віртуальному осцилографі. На рис. 1.22 наведений один з варіантів побудови сигналу.

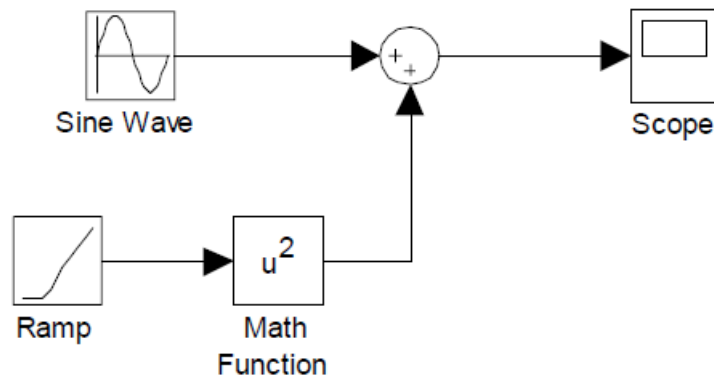


Рис. 1.22. Модель сигналу $x(t) = 0,5 * \sin \pi t + t^2$

Сигнал $0.5\sin(\pi t)$ заданий у параметрах блоку Sine Wave (рис. 1.23).

Для сигналу t^2 використовувалися два блоки - блок лінійного сигналу і блок математичної функції, де була обрана функція зведення в квадрат.

Результати роботи виведені на екран віртуального осцилографа (рис. 1.24).

Source Block Parameters: Sine Wave

Sine Wave

Output a sine wave:

$$Q(t) = \text{Amp} \cdot \sin(\text{Freq} \cdot t + \text{Phase}) + \text{Bias}$$

Sine type determines the computational technique used. The parameters in the two types are related through:

$$\text{Samples per period} = 2\pi / (\text{Frequency} \cdot \text{Sample time})$$

$$\text{Number of offset samples} = \text{Phase} \cdot \text{Samples per period} / (2\pi)$$

Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur.

Parameters

Sine type: **Time based**

Time (t): **Use simulation time**

Amplitude: **0.5**

Bias: **0**

Frequency (rad/sec): **pi**

Phase (rad): **0**

Sample time: **0**

☒ Interpret vector parameters as 1-D

OK Cancel Help

Рис. 1.23. Параметры блока Sine Wave.

Інтервал моделювання заданий в межах від 0 до 2 у вікні меню Simulation / Configuration Parameters.

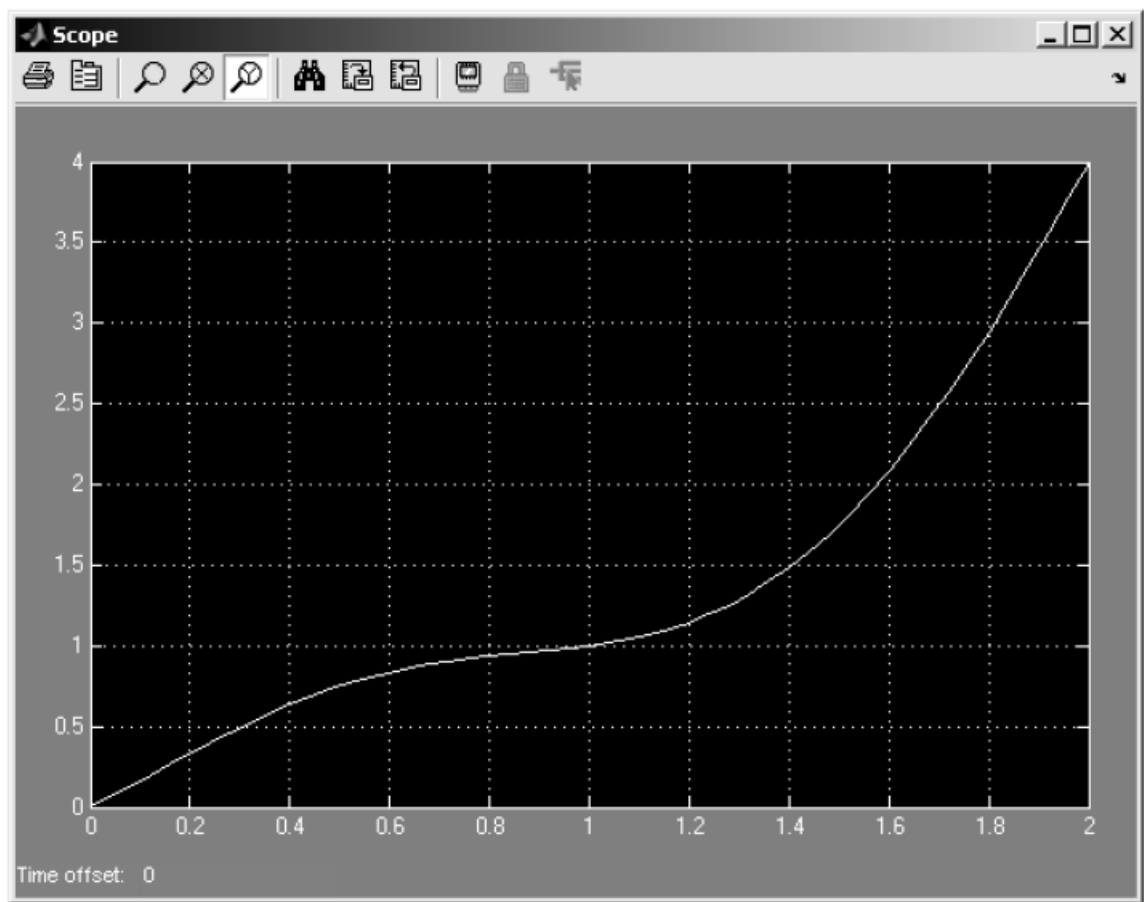


Рис. 1.24. Сигнал $x(t) = 0,5 * \sin \pi t + t^2$ на екрані віртуального осцилографа.

Комп'ютерний практикум № 1
ЗАГАЛЬНІ ВІДОМОСТІ ПРО ПРОГРАМНИЙ КОМПЛЕКС SIMULINK


Завдання

1. Створити новий файл моделі.
2. Розташувати у вікні моделі блоки Sine (розділ Sources), Scope, To Workspace (розділ Sinks).
3. З'єднати елементи схеми: блок Sine та підключити до блоків Scope та To Workspace. У блоці To Workspace задати формат array для експортуємих даних; ім'я змінної задати sin1.
4. Зберегти схему у вигляді файлу на диску у **своїй** папці.
5. Додати до моделі текстові написи: група, ПІБ, номер комп'ютерного практикуму.
6. Скопіювати блоки Sine, Scope та To Workspace, вставити їх у цю ж модель та з'єднати як у пункті 3 (ім'я змінної задати sin2).
7. Додати до моделі блоки Sum (розділ Math Operations), Scope та To Workspace (задати формат array для експортуємих даних; ім'я змінної задати sin1_plus_sin2). З'єднати усі блоки таким чином, щоб на першому осцилографі можна було побачити графік першої синусоїди (а її значення потрапляли на перший блок To Workspace), на другому – другої синусоїди (а її значення - на другий блок To Workspace), на третьому – суми синусоїд (а її значення - на третій блок To Workspace).
8. Перемістити блоки моделі таким чином, щоб вони були зручно розташовані у вікні моделі.
9. Задати параметри для двох функцій sin згідно з номером варіанту (див. табл. 1.1).

$$f(x) = A\sin(2\pi f + \varphi) + a$$

де A – амплітуда синусоїдального сигналу, f – його частота, φ - фаза, a – зсув.

Sample time для першої синусоїди задати 0.02, для другої – 0.

10. Запустити модель на виконання, вказавши Simulation time – 10.
11. Переконайтесь по графіку, що амплітуда, частота, фаза та зсув відповідають тим, що ви ввели згідно свого варіанту (для цього можна збільшити зображення графіку за допомогою інструменту ). Якщо сигнал не вміщується у вікні Scope,

виберіть команду Axes properties, натиснувши праву кнопку миші на вікні Scope, та змініть максимальне та мінімальне значення по осі у.

12. Збільшити зображення графіку першої синусоїди та переконатись, що параметр Sample time відповідає тому, що введено в блоці Source Block Parameters: Sine Wave.

13. Перейти до вікна Matlab, перевірити наявність матриць, які були отримані за допомогою трьох блоків To Workspace (ці матриці знаходяться у вікні Workspace Матлабу; якщо це вікно закрите, зайдіть у меню Desktop (або Вікно) та активуйте пункт Workspace (Робочий простір)). Подумати, що за змінна tout з'явилася у вікні Workspace.

14. Побудувати в середовищі Matlab (**в m-файлі**) три графіки для sin1, sin2 та sin1_plus_sin2 на одній figure за допомогою команди subplot. Підписати графіки та вісі.

15. Порівняти ці графіки з графіками на екранах блоків Scope.

16. Зберегти усі малюнки, m-файл та модель у **своїй папці** (вони можуть знадобитися для наступних робіт).

Варіанти завдань

Таблиця 1.1

Варіанти	A1	f1	φ_1	a1
	A2	f2	φ_2	a2
1	3	3	0	0.5
	1	0.15	π	0
2	4	4	π	0.3
	1	0.2	π	0
3	6	6	π	1
	2	0.3	0	2
4	4	3.5	2π	-2
	2	0.2	π	1
5	4	5	3π	-0.5
	1	0.25	0	1
6	5	7	4π	0.6
	2	0.35	π	-2

7	4 1	4 0.2	0 2pi	0.5 1
8	3 1	3 0.15	0 pi	1 -1
9	4 1	4 0.2	pi pi	2 -0.5
10	5 2	6 0.3	pi 0	0.5 0
11	3.5 2	3.5 0.2	2pi pi	0.3 0
12	5 1	5 0.25	3pi 0	1 2
13	7 2	7 0.35	4pi pi	-2 1
14	4 1	4 0.2	0 2pi	-0.5 1
15	4.5	4.5 0.2	pi pi	0.6 -2
16	5 2	2 0.1	pi 0	0.5 1
17	5.5 1	5.5 0.3	2pi pi	1 -1
18	5 2	6 2	3pi 0	2 -0.5

Література:

1. Ануфриев И.Е., Смирнов А.Б., Смирнова Е.Н. MATLAB 7. – СПб.: БХВ–Петербург, 2005. – 1104 с.
2. Данилов А.И. Компьютерный практикум по курсу «Теория управления». SIMULINK – моделирование в среде MATLAB. – М.: МГУИЭ, 2002, 128 с.
3. Дьяконов В.П. MATLAB 6/6.1/6.5 + Simulink 4/5. Основы применения. Полное руководство пользователя. – М.: СОЛОН-Пресс. -2002. – 768 с.
4. Дэбни Дж., Хартман Т. Simulink 4. Секреты мастерства. – М.: БИНОМ. Лаборатория знаний, 2003. – 403 с.

5. Черных И.В. SIMULINK: среда создания инженерных приложений. – М.: ДИАЛОГ-МИФИ, 2003. – 496 с.

Лекція 2. Бібліотека блоків Simulink: Sources та Math Operations

2.1. Sources - джерела сигналів

2.1.1. Джерело постійного сигналу Constant

Призначення:

Задає постійний за рівнем сигнал.

Параметри:

1. Constant value - Постійна величина.
2. Interpret vector parameters as 1-D - Інтерпретувати вектор параметрів як одновимірний (при встановленому прапорці). Даний параметр зустрічається у більшості блоків бібліотеки Simulink.

Значення константи може бути дійсним або комплексним числом, обчислюваним виразом, вектором або матрицею.

Рис. 2.1.1 ілюструє застосування цього джерела і вимір його вихідного сигналу за допомогою цифрового індикатора Display.

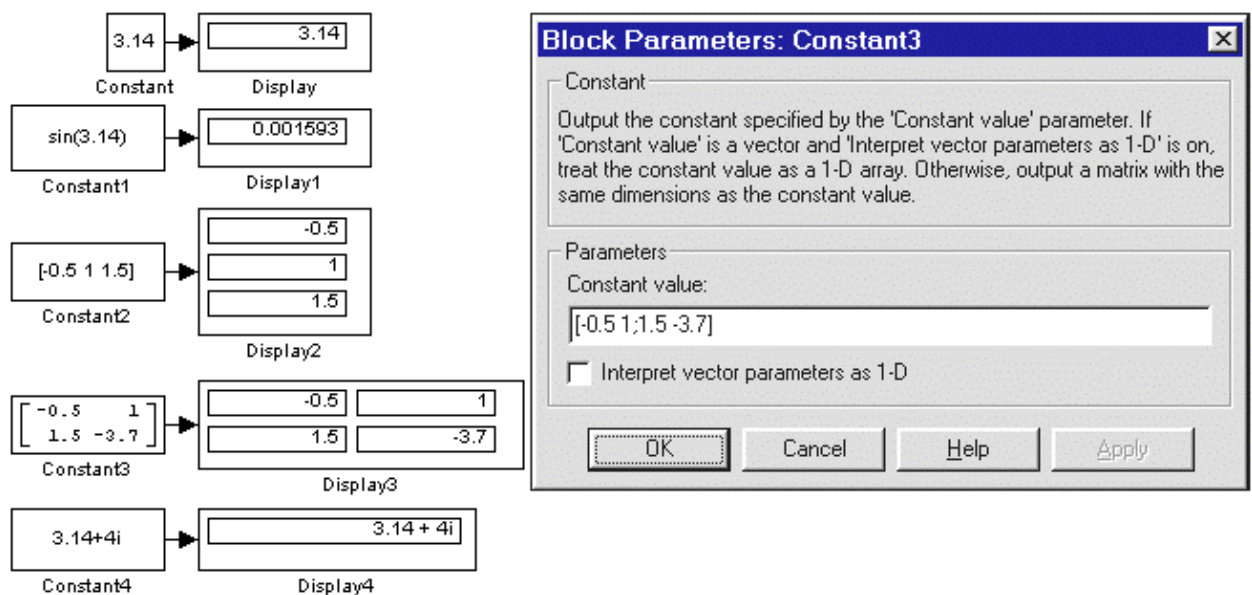


Рис. 2.1.1. Джерело постійного впливу Constant

2.1.2. Джерело синусоїдального сигналу Sine Wave

Призначення:

Формує синусоїдальний сигнал із заданою частотою, амплітудою, фазою і зміщенням.

Для формування вихідного сигналу блоком можуть використовуватися два алгоритми. Вид алгоритму визначається параметром Sine Type (спосіб формування сигналу):

- Time-based - За поточним часу.
- Sample-based - За величиною кроку модельного часу.

2.1.2.1. Формування вихідного сигналу за поточним значенням часу для безперервних систем

Вихідний сигнал джерела в цьому режимі відповідає виразу:

$$y = \text{Amplitude} * \sin(\text{frequency} * \text{time} + \text{phase}) + \text{bias}.$$

Параметри:

1. Amplitude - Амплітуда.
2. Bias - Постійна складова сигналу.
3. Frequency (rads / sec) - Частота (рад / с).
4. Phase (rads) - Початкова фаза (рад).
5. Sample time - Крок модельного часу. Використовується для узгодження роботи джерела та інших компонентів моделі в часі. Параметр може приймати наступні значення:

0 (за замовчуванням) - Використовується при моделюванні безперервних систем.

> 0 (позитивне значення) - Здається при моделюванні дискретних систем. У цьому випадку крок модельного часу можна інтерпретувати як крок квантування за часом вихідного сигналу.

-1 - Крок модельного часу встановлюється таким же, як і в попередньому блоці, тобто блоці, звідки приходить сигнал в даний блок.

Цей параметр може задаватися для більшості блоків бібліотеки Simulink.

2.1.2.2. Формування вихідного сигналу за поточним значенням часу для дискретних систем

Алгоритм визначення значення вихідного сигналу джерела для кожного наступного кроку розрахунку визначається виразом (в матричній формі):

$$\begin{bmatrix} \sin(t + \Delta t) \\ \cos(t + \Delta t) \end{bmatrix} = \begin{bmatrix} \cos(\Delta t) & \sin(\Delta t) \\ -\sin(\Delta t) & \cos(\Delta t) \end{bmatrix} \begin{bmatrix} \sin(t) \\ \cos(t) \end{bmatrix},$$

де Δt - постійна величина, що дорівнює значенню Sample time.

У даному режимі помилка округлення для великих значень часу також зменшує точність розрахунку.

2.1.2.3. Формування вихідного сигналу за величиною модельного часу і кількості розрахункових кроків на один період

Вихідний сигнал джерела в цьому режимі відповідає виразу:

$$y = \text{Amplitude} * \sin [(k + \text{Number of offset samples}) / \text{Samples per period}] + \text{bias},$$

де k - номер поточного кроку розрахунку.

Параметри:

1. Amplitude - Амплітуда.

2. Bias - Постійна складова сигналу.

3. Samples per period - Кількість розрахункових кроків на один період синусоїдального сигналу:

$$\text{Samples per period} = 2\pi / (\text{frequency} * \text{Sample time})$$

4. Number of offset samples - Початкова фаза сигналу. Здається кількістю кроків модельного часу:

$$\text{Number of offset samples} = \text{Phase} * \text{Samples per period} / (2\pi).$$

5. Sample time - Крок модельного часу.

У даному режимі помилка округлення не накопичується, оскільки Simulink починає відлік номера поточного кроку з нуля для кожного періоду.

На рис. 2.1.2 показано застосування блоку з різними значеннями кроку модельного часу (Sample time = 0 для блоку Sine Wave 1 і Sample time = 0.1 для блоку Sine Wave 2). Для відображення графіків вихідних сигналів в моделі використаний віртуальний осцилограф (Scope).

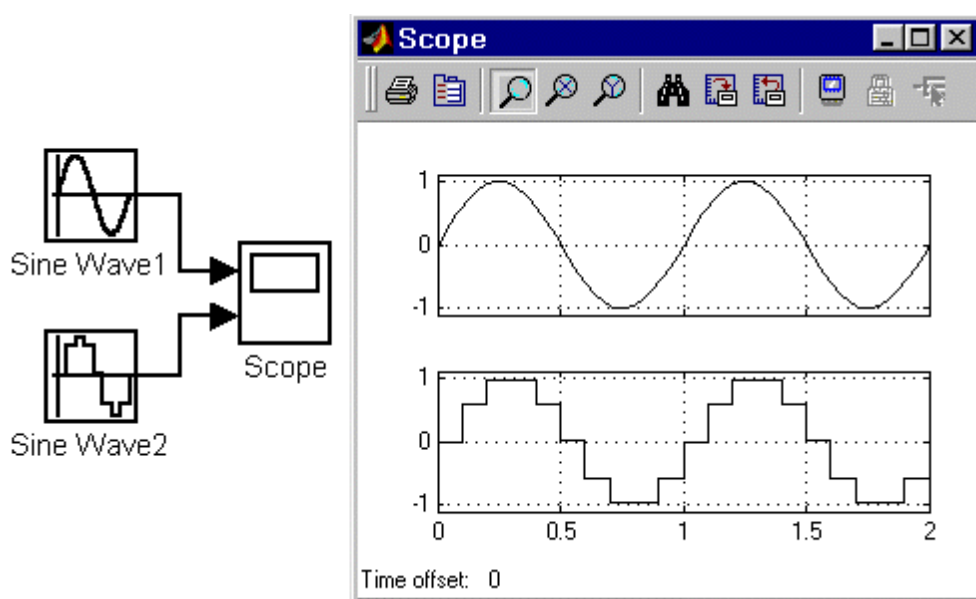


Рис. 2.1.2. Блок Sine Wave

2.1.3. Джерело лінійно змінюючогося впливу Ramp

Призначення: Формує лінійний сигнал виду $y = \text{Slope} * \text{time} + \text{Initial value}$.

Параметри:

1. Slope - Швидкість зміни вихідного сигналу.
2. Start time - Час початку формування сигналу.
3. Initial value - Початковий рівень сигналу на виході блоку.

На рис. 2.1.3. показано використання даного блоку.

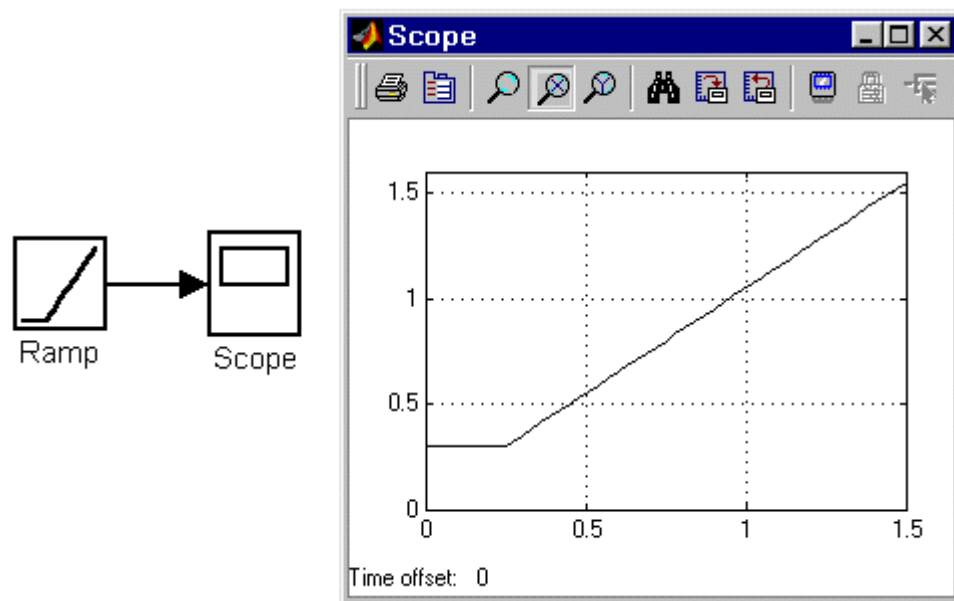


Рис. 2.1.3. Блок Ramp

2.1.4. Генератор ступінчастого сигналу Step

Призначення: Формує ступінчастий сигнал.

Параметри:

1. Step time - Час настання перепаду сигналу (с).
2. Initial value - Початкове значення сигналу.
3. Final value - Кінцеве значення сигналу.

Перепад може бути як в більшу сторону (кінцеве значення більше ніж початкове), так і в меншу (кінцеве значення менше ніж початкове). Значення початкового та кінцевого рівнів можуть бути не тільки позитивними, але й негативними (наприклад, зміна сигналу з рівня -5 до рівня -3).

На рис. 2.1.4. показано використання генератора ступінчастого сигналу.

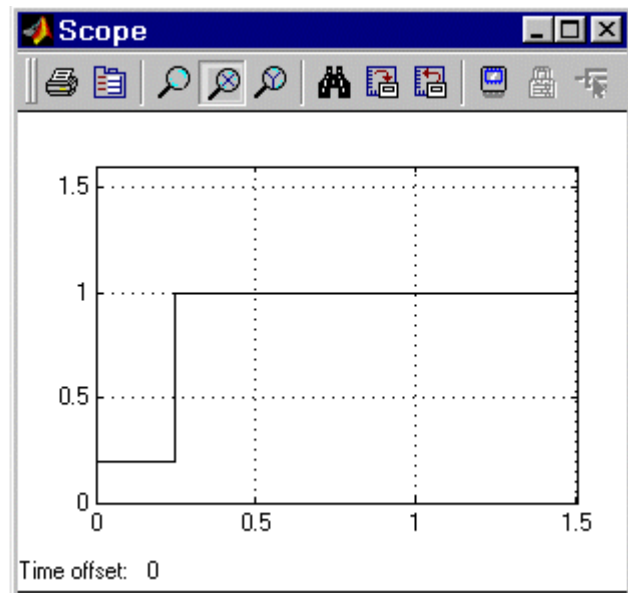
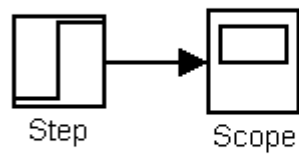


Рис. 2.1.4. Блок Step

2.1.5. Генератор сигналів Signal Generator

Призначення: Формує один з чотирьох видів періодичних сигналів:

1. sine - Синусоїдальний сигнал.
2. square - Прямокутний сигнал.
3. sawtooth - Пілкоподібний сигнал.
4. random - Випадковий сигнал.

Параметри:

1. Wave form - Вид сигналу.
2. Amplitude - Амплітуда сигналу.
3. Frequency - Частота (рад / с).
4. Units - Одиниці виміру частоти. Може приймати два значення:
 - Hertz - Гц.
 - Rad / sec - рад / с.

На рис. 2.1.5. показано застосування цього джерела при моделюванні прямокутного сигналу.

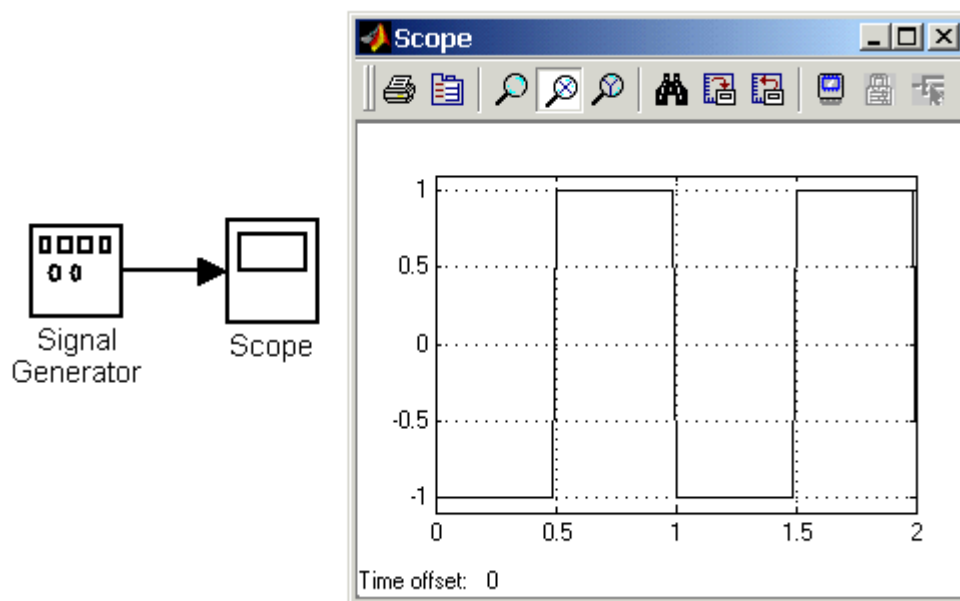


Рис. 2.1.5. Блок генератора сигналів

2.1.6. Джерело випадкового сигналу з рівномірним розподілом Uniform Random Number

Призначення: Формування випадкового сигналу з рівномірним розподілом.

Параметри:

1. Minimum - Мінімальний рівень сигналу.
2. Maximum - Максимальний рівень сигналу.
3. Initial seed - Початкове значення.

Приклад використання блоку і графік його вихідного сигналу представлений на рис. 2.1.6.

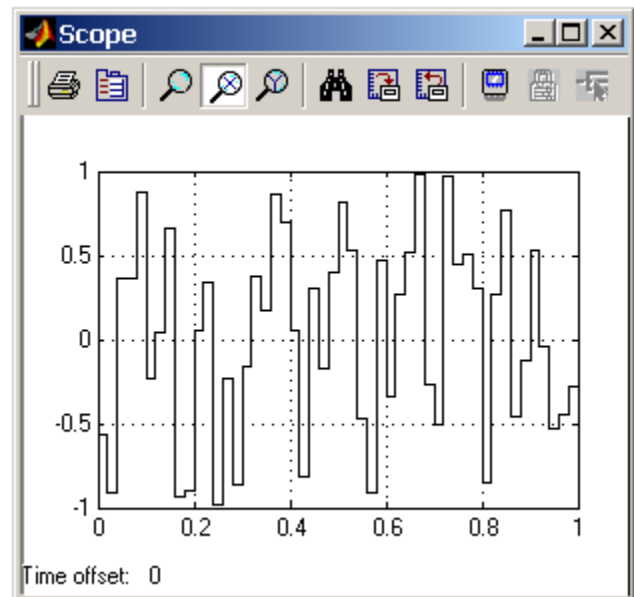
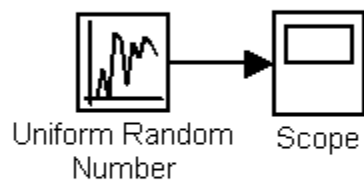


Рис. 2.1.6. Джерело випадкового сигналу з рівномірним розподілом

2.1.7. Джерело випадкового сигналу з нормальним розподілом Random Number

Призначення: Формування випадкового сигналу з нормальним розподілом рівня сигналу.

Параметри:

1. Mean - Середнє значення сигналу
2. Variance- Дисперсія (середньоквадратичне відхилення).
3. Initial seed - Початкове значення.

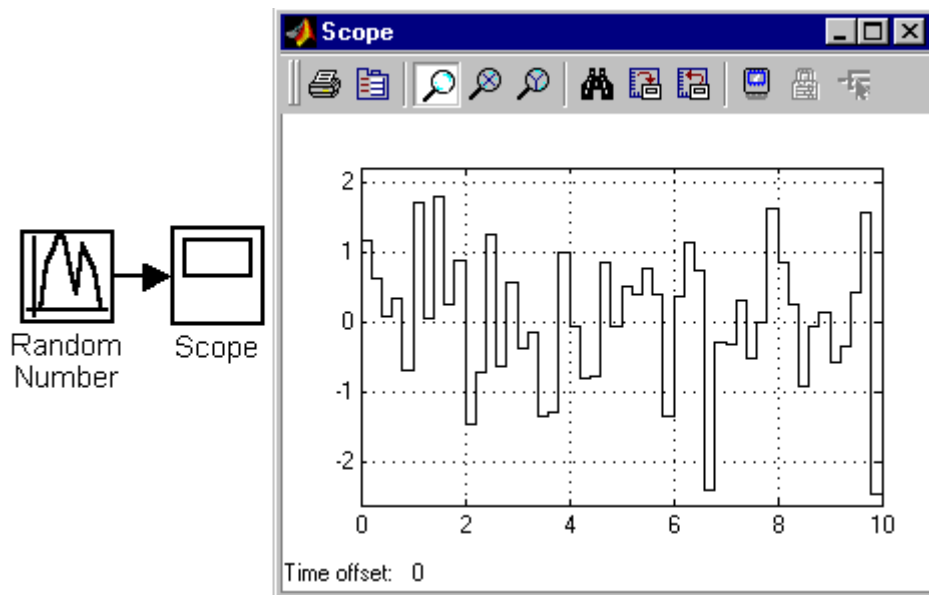


Рис. 2.1.7. Джерело випадкового сигналу з нормальним розподілом

2.1.8. Джерело імпульсного сигналу Pulse Generator

Призначення: Формування прямокутних імпульсів.

Параметри:

1. Pulse Type - Спосіб формування сигналу. Може приймати два значення:
 - Time-based - За поточним часу.
 - Sample-based - За величиною модельного часу і кількості розрахункових кроків.
2. Amplitude - Амплітуда.
3. Period - Період. Здається в секундах для Time-based Pulse Type або в кроках модельного часу для Sample-based Pulse Type.
4. Pulse width - Ширина імпульсів. Здається в у% по відношенню до періоду для Time-based Pulse Type або в кроках модельного часу для Sample-based Pulse Type.
5. Phase delay - Фазова затримка. Здається в секундах для Time-based Pulse Type або в кроках модельного часу для Sample-based Pulse Type.
6. Sample time - Крок модельного часу. Здається для Sample-based Pulse Type.

Приклад використання Pulse Generator показаний на рис. 2.1.8.

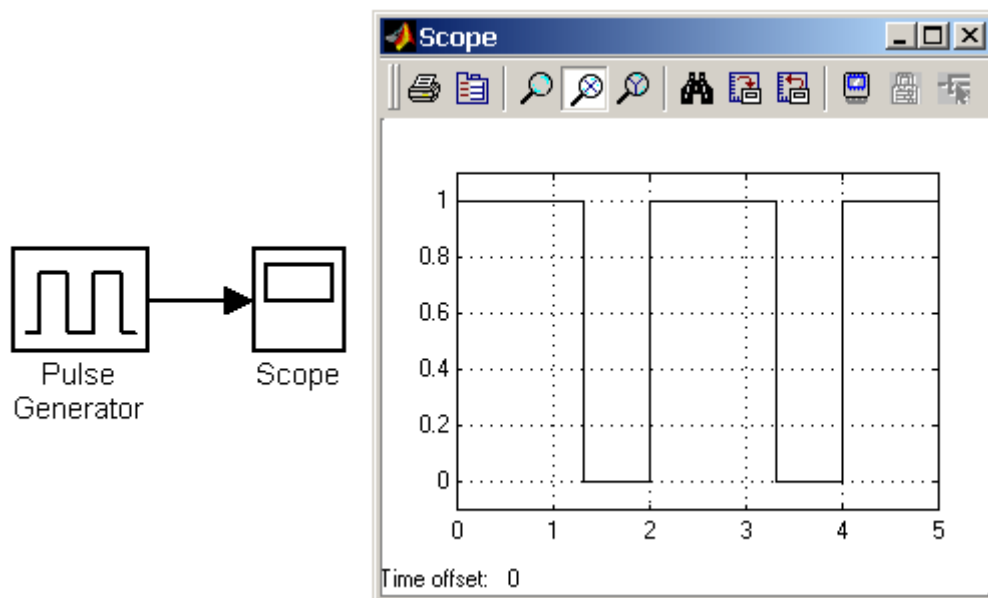


Рис. 2.1.8. Джерело прямокутних імпульсів

2.1.9. Генератор лінійно-змінюючої частоти Chirp Generator

Призначення: Формування синусоїдальних коливань, частота яких лінійно змінюється.

Параметри:

1. Initial frequency - Початкова частота (Гц);
2. Target time - Час зміни частоти (с);
3. Frequency at target time - Кінцеве значення частоти (Гц).

Приклад використання блоку показаний на рис. 2.1.9.

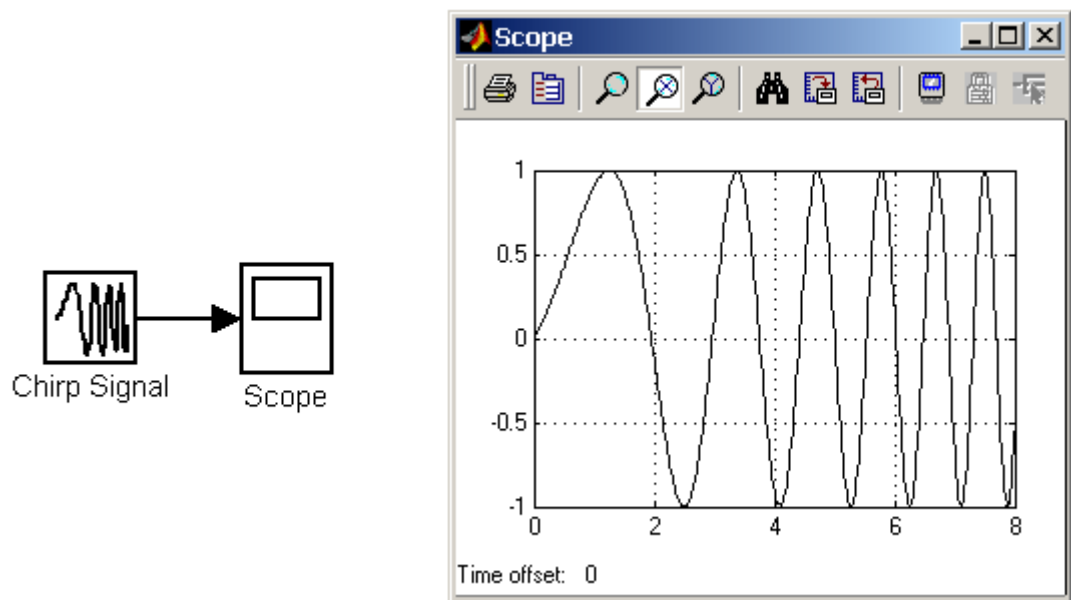


Рис. 2.1.9. Генератор лінійно-змінюється частоти

2.1.10. Генератор білого шуму Band-Limited White Noise

Призначення: Створює сигнал заданої потужності, рівномірно розподілений по частоті.

Параметри:

1. Noise Power - Потужність шуму.
2. Sample Time - Модельний час.
3. Seed - Число, необхідне для ініціалізації генератора випадкових чисел.

Рис. 2.1.10 показує роботу цього генератора.

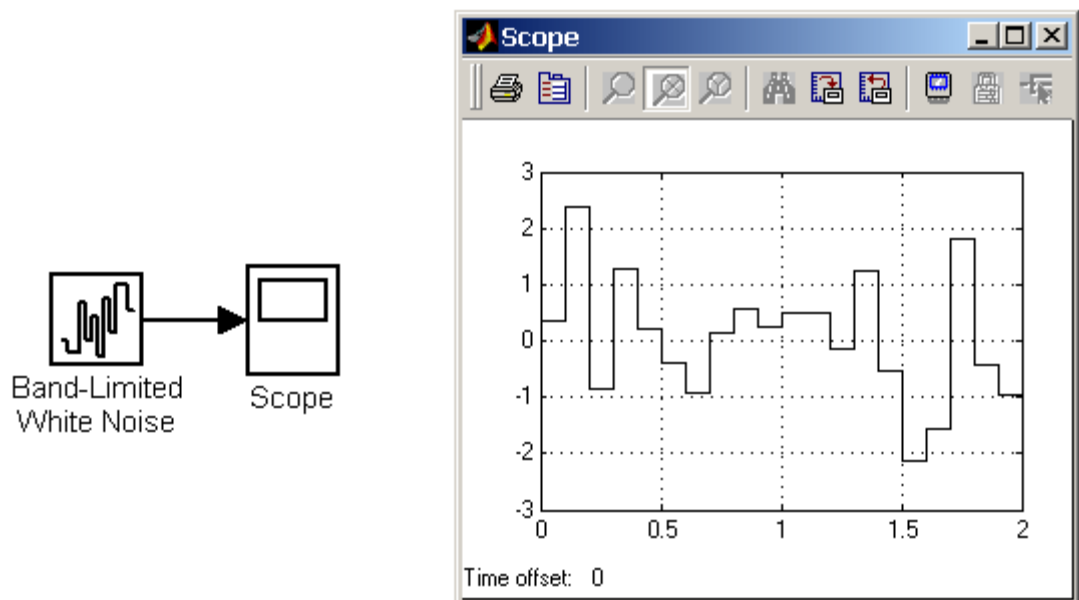


Рис. 2.1.10. Генератор білого шуму

2.1.11. Джерело тимчасового сигналу Clock

Призначення: Формує сигнал, величина якого на кожному кроці розрахунку дорівнює поточному часу моделювання.

Параметри

1. Decimation - Крок, з яким оновлюються показання часу на зображенні джерела (в тому випадку, якщо встановлений прапорець параметра Display time). Параметр задається як кількість кроків розрахунку. Наприклад, якщо крок розрахунку моделі у вікні діалогу Simulation parameters встановлений рівним 0.01 с, а параметр Decimation блоку Clock заданий рівним 1000, то оновлення показань часу буде проводитися кожні 10 з модельного часу.

2. Display time - Відображення значення часу в блоці джерела.

На рис. 2.1.11 показаний приклад роботи даного джерела.

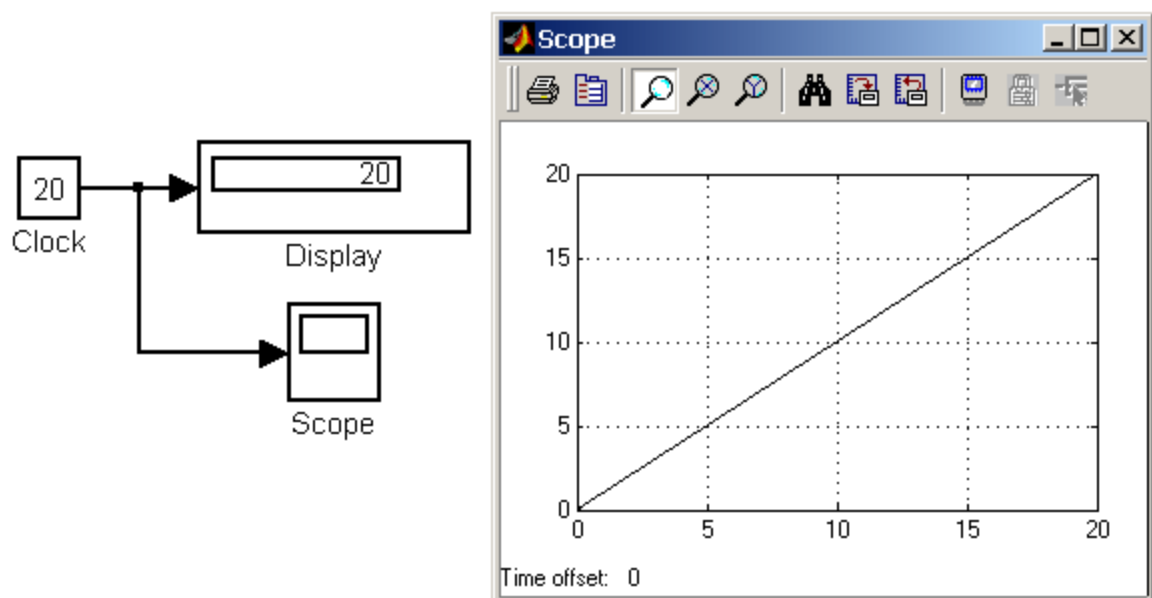


Рис. 2.1.11. Джерело тимчасового сигналу

2.1.12. Цифрове джерело часу Digital Clock

Призначення: Формує дискретний часовий сигнал.

Параметр:

Sample time - Крок модельного часу (с).

На рис. 2.1.12 показана робота джерела Digital Clock.

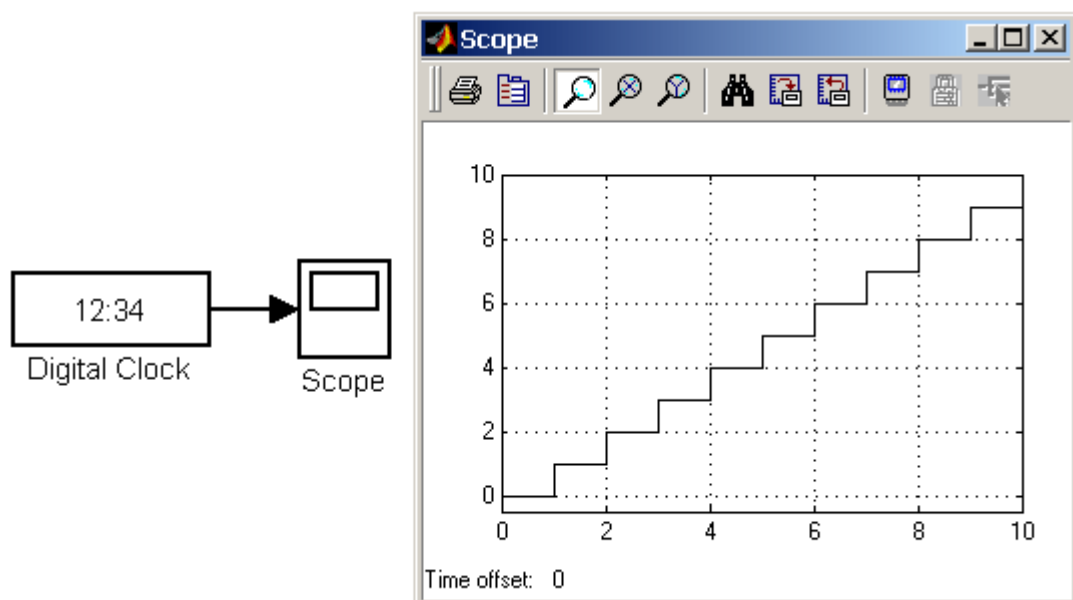


Рис. 2.1.12. Цифрове джерело тимчасового сигналу

2.1.13. Блок зчитування даних з файлу From File

Призначення: Отримання даних із зовнішнього файлу.

Параметри:

1. File Name - Ім'я файла з даними.
2. Sample time - Крок зміни вихідного сигналу блоку.

Дані у файлі повинні бути представлені у вигляді матриці:

$$\begin{bmatrix} t_1 & t_2 & \dots & t_{final} \\ u1_1 & u1_2 & \dots & u1_{final} \\ \dots & \dots & \dots & \dots \\ un_1 & un_2 & \dots & un_{final} \end{bmatrix}.$$

Матриця повинна складатися, як мінімум, з двох рядків. Значення часу записані в першому рядку матриці, а в інших рядках знаходяться значення сигналів, відповідні даними моментів часу. Значення часу повинні бути записані в зростаючому порядку. Вихідний сигнал блоку містить тільки значення сигналів, а значення часу в ньому відсутні. Якщо крок розрахунку поточної моделі не збігається з отсчетами часу у файлі даних, то Simulink виконує лінійну інтерполяцію даних.

Файл даних (mat-файл), з якого зчитуються значення, не є текстовим. Структура файлу докладно описана в довідковій системі MATLAB. Користувачам Simulink найзручніше створювати mat-файл за допомогою блоку To File (бібліотека Sinks). На рис. 2.1.13 показаний приклад використання даного блоку. З файлу data.mat зчитуються значення синусоїдального сигналу.

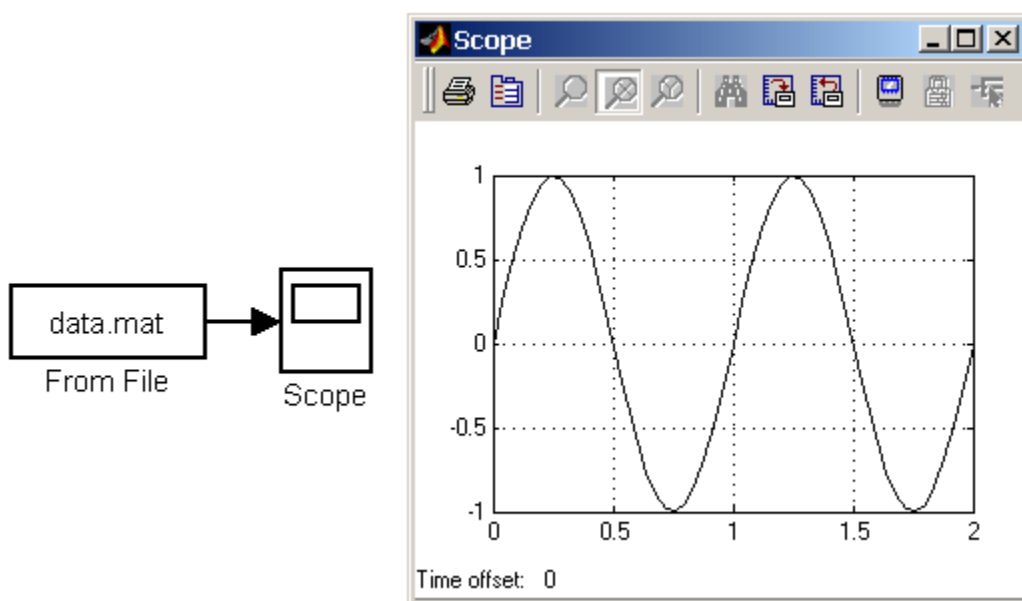


Рис. 2.1.13. Блок From File

2.1.14. Блок зчитування даних з робочого простору From Workspace

Призначення: Отримання даних з робочого простору MATLAB.

Параметри:

1. Data - Ім'я змінної (матриці або структури) містить дані.
2. Sample time - Крок зміни вихідного сигналу блоку.
3. Interpolate data - Інтерполяція даних для значень модельного часу не збігаються зі значеннями у змінній Data.
4. Form output after final data value by - Вид вихідного сигналу по закінченні значень часу у змінній Data:
 - Extrapolate - Лінійна екстраполяція сигналів.
 - SettingToZero - Нульові значення сигналів.
 - HoldingFinalValue - Вихідні значення сигналів дорівнюють останнім значенням.
 - CyclicRepetition - Циклічне повторення значень сигналів. Даний варіант може використовуватися, тільки якщо змінна Data має формат Structure without time.

На рис. 2.1.14 показаний приклад використання даного блоку. Дані в змінну simin робочої області MATLAB завантажуються з файлу за допомогою блоку Read data.

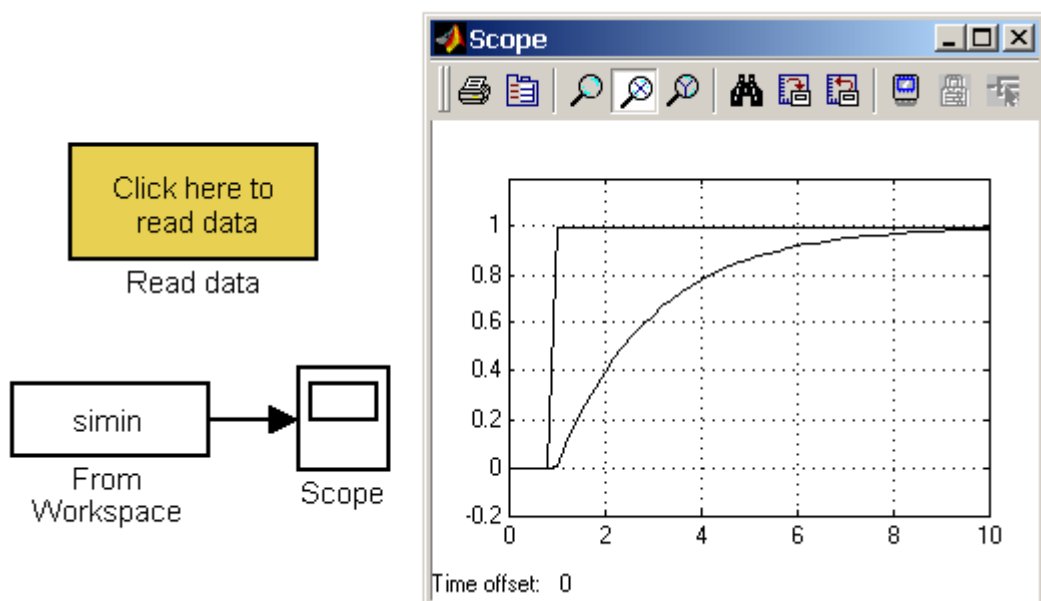


Рис. 2.1.14. Блок From File

2.1.15. Блок сигналу нульового рівня Ground

Призначення: Формування сигналу нульового рівня.

Параметри:

Немає.

Якщо який-небудь вхід блоку в моделі не підключений, то при виконанні моделювання в головному вікні MATLAB з'являється попередження. Для усунення цього на непідключений вхід блоку можна подати сигнал з блоку Ground.

На рис. 2.1.15 дані приклади використання блоку. У першому випадку сигнал з блоку Ground потрапляє на один з входів суматора, а в другому на один з входів блоку множення. Показання блоків Display підтверджують, що виробляється блоком Ground сигнал має нульове значення. З малюнка також видно, що тип вихідного сигналу блоку встановлюється автоматично, відповідно до типів сигналів, що подаються на інші входи блоків (в даному випадку - на входи блоків Sum і Product).

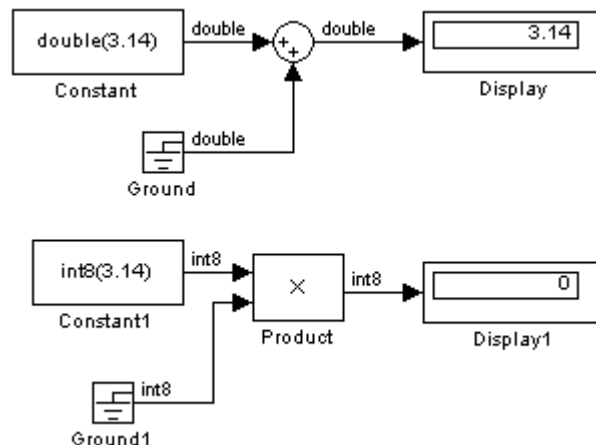


Рис. 2.1.15. Застосування блоку Ground

2.1.16. Блок періодичного сигналу Repeating Sequence

Призначення: Формування періодичного сигналу.

Параметри:

1. Time values - Вектор значень модельного часу.
2. Output values - Вектор значень сигналу для моментів часу заданих вектором Time values.

Блок виконує лінійну інтерполяцію вихідного сигналу для моментів часу не збігаються зі значеннями заданими вектором Time values. На рис. 2.1.16 показаний приклад використання блоку для формування пилоподібного сигналу. Значення

модельного часу задані вектором [0 3], а значення вихідного сигналу вектором [0 2].

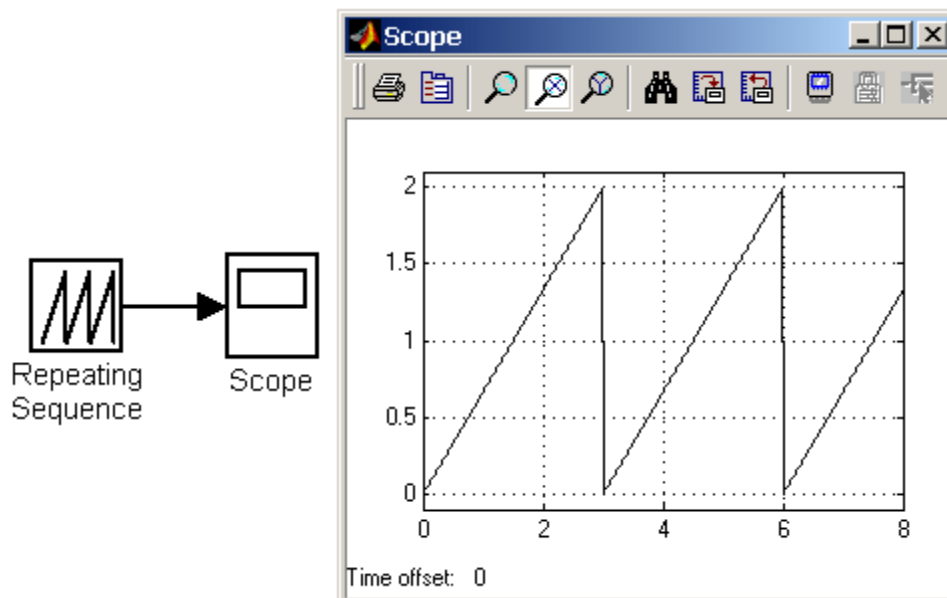


Рис. 2.1.16. Використання блоку Repeating Sequence

2.1.17. Блок вхідного порту Inport

Призначення: Створює вхідний порт для підсистеми або моделі верхнього рівня ієрархії.

Параметри:

- Port number - Номер порту.
- Port dimensions - Розмірність вхідного сигналу. Якщо цей параметр дорівнює -1, то розмірність вхідного сигналу буде визначатися автоматично.
- Sample time - Крок модельного часу.
- Data type - Тип даних вхідного сигналу: auto, double, single, int8, uint8, int16, uint16, int32, uint32 або boolean.
- Signal type - Тип вхідного сигналу:
 1. auto - Автоматичне визначення типу.
 2. real-дійсний сигнал.
 3. complex- Комплексний сигнал.
- Interpolate data (прапорець) - інтерпольований вхідний сигнал. У випадку, якщо тимчасові відліки вхідного сигналу зчитуваного з робочої області MATLAB не збігаються з модельним часом, то блок буде виконувати інтерполяцію вхідного сигналу. При використанні блоку Inport в підсистемі даний параметр не доступний.

2.1.17.1. Використання блоку Inport в підсистемах

Блоки Inport підсистеми є її входами. Сигнал, що подається на вхідний порт підсистеми через блок Inport, передається всередину підсистеми. Назва вхідного порту буде показано на зображенні підсистеми як мітка порту.

При створенні підсистем і додаванні блоку Inport в підсистему Simulink використовує наступні правила:

1. При створенні підсистеми за допомогою команди Edit / Create subsystem вхідні порти створюються і нумеруються автоматично починаючи з 1.
2. Якщо в підсистему додається новий блок Inport, то йому присвоюється наступний по порядку номер.
3. Якщо будь-якої блок Inport віддаляється, то інші порти перейменовуються таким чином, щоб послідовність номерів портів була безперервною.
4. Якщо в послідовності номерів портів є розрив, то при виконанні моделювання Simulink видасть повідомлення про помилку і зупинить розрахунок. У цьому випадку необхідно вручну перейменувати порти таким чином, щоб послідовність номерів портів не порушувалася.

На рис. 2.1.17 показана модель, що використовує підсистему і схема цієї підсистеми.

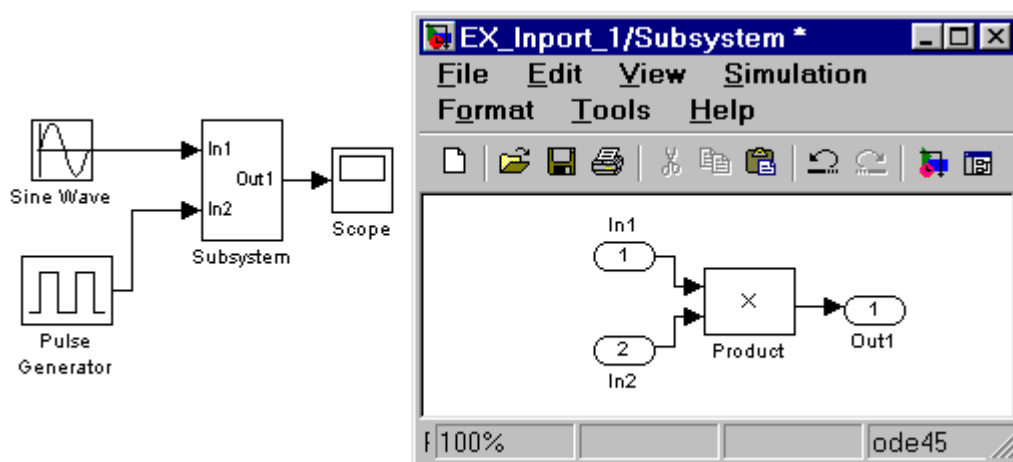


Рис. 2.1.17. Використання блоку Inport в підсистемі

2.1.17.2. Використання блоку Inport в моделі верхнього рівня

Вхідний порт в системі верхнього рівня використовується для передачі сигналу з робочої області MATLAB в модель.

Для передачі сигналу з робочого простір MATLAB потрібно не тільки встановити в моделі вхідний порт, але і виконати установку параметрів введення

на вкладці Workspace I / О вікна діалогу Simulation parameters ... (повинен бути встановлений прапорець для параметра Input і задано ім'я змінної, яка містить вхідні дані). Тип даних, що вводяться: Array (масив), Structure (структура) або Structure with time (структура з полем "час") задається на цій же вкладці.

На рис. 2.1.18 показана модель, що зчитує вхідний сигнал з робочого простору MATLAB. Підсистема Load Data забезпечує введення даних з файлу в робочу область MATLAB.

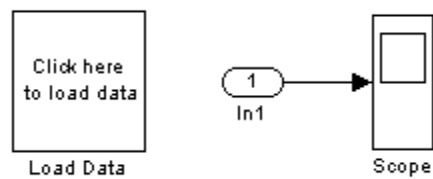


Рис. 2.1.18. Модель, що зчитує вхідний сигнал з робочого простору MATLAB за допомогою блоку Input.

2.2. Math - блоки математичних операцій

2.2.1. Блок обчислення модуля Abs

Призначення: Виконує обчислення абсолютного значення величини сигналу.

Параметри:

Saturate on integer overflow (прапорець) - Придушувати переповнення цілого.

При встановленому прапорці обмеження сигналів цілого типу виконується коректно.

Приклад використання блоку Abs, що обчислює модуль поточного значення синусоїдального сигналу, показаний на рис. 2.2.1.

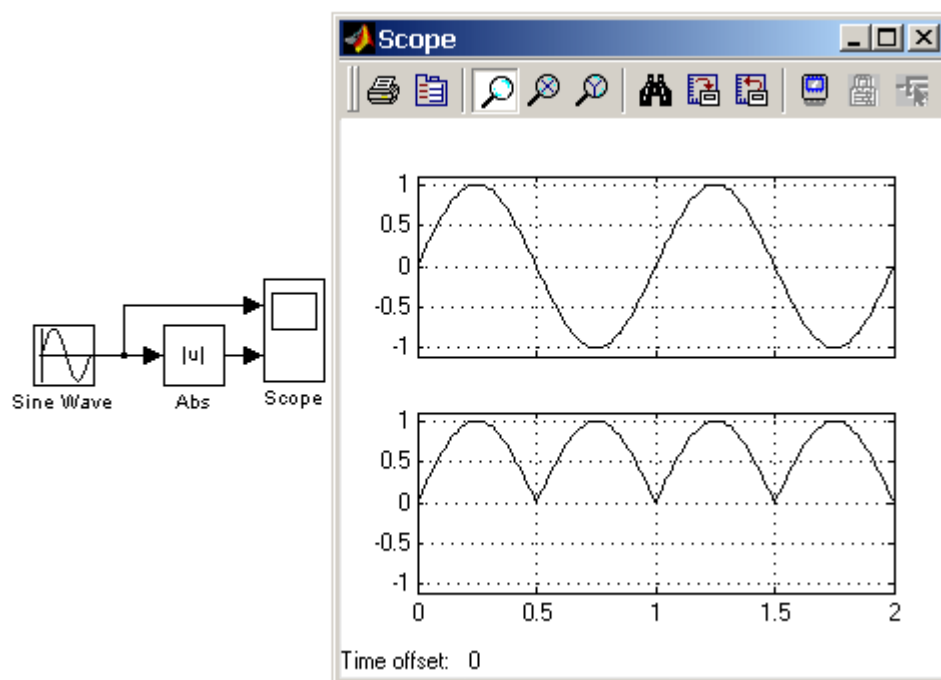


Рис. 2.2.1. Приклад використання блоку Abs

Блок Abs може використовуватися також для обчислення модуля сигналу комплексного типу. На рис. 2.2.2 показаний приклад обчислення модуля комплексного сигналу виду:

$$u = \cos(\omega \cdot t) + i \cdot \sin(\omega \cdot t)$$

Модуль цього сигналу (як і слід було очікувати) дорівнює 1 для будь-якого моменту часу.

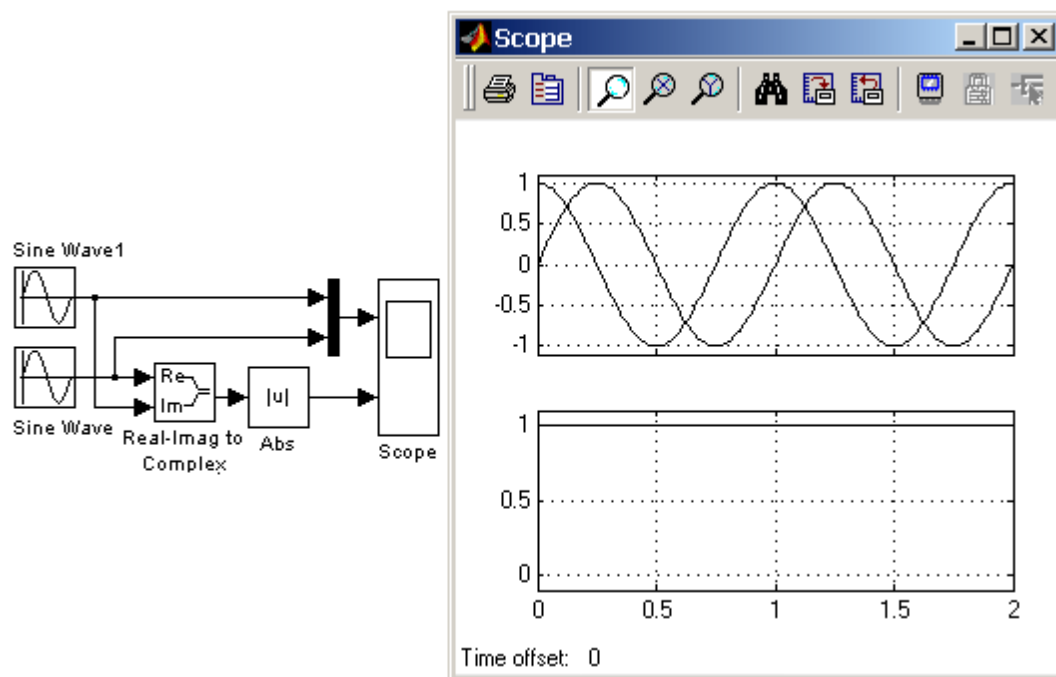


Рис. 2.2.2. Приклад використання блоку Abs для обчислення модуля комплексного сигналу

2.2.2. Блок обчислення суми Sum

Призначення: Виконує обчислення суми поточних значень сигналів.

Параметри:

Icon shape - Форма блоку. Вибирається зі списку.

- Round - коло,
- Rectangular - прямокутник.

List of sign - Список знаків. У списку можна використовувати такі знаки:

+ (Плюс), - (мінус) і | (роздільник знаків).

Saturate on integer overflow (прапорець) - Придушувати переповнення цілого.

При встановленому прапорці обмеження сигналів цілого типу виконується коректно.

Кількість входів і операція (додавання чи віднімання) визначається списком знаків параметра List of sign, при цьому мітки входів позначаються відповідними знаками. У параметрі List of sign можна також вказати число входів блоку. У цьому випадку всі входи будуть сумуючими.

Якщо кількість входів блоку перевищує 3, то зручніше використовувати блок Sum прямокутної форми.

Блок може використовуватися для підсумовування скалярних, векторних або матричних сигналів. Типи сумуємих сигналів повинні збігатися. Не можна, наприклад, подати на один і той же сумуючий блок сигнали цілого і дійсного типів.

Якщо кількість входів блоку більше, ніж один, то блок виконує поелементні операції над векторними і матричними сигналами. При цьому кількість елементів в матриці або векторі має бути однаковим.

Якщо в якості списку знаків вказати цифру 1 (один вхід), то блок можна використовувати для визначення суми елементів вектора.

Приклади використання блоку Sum показані на 2.2.3.

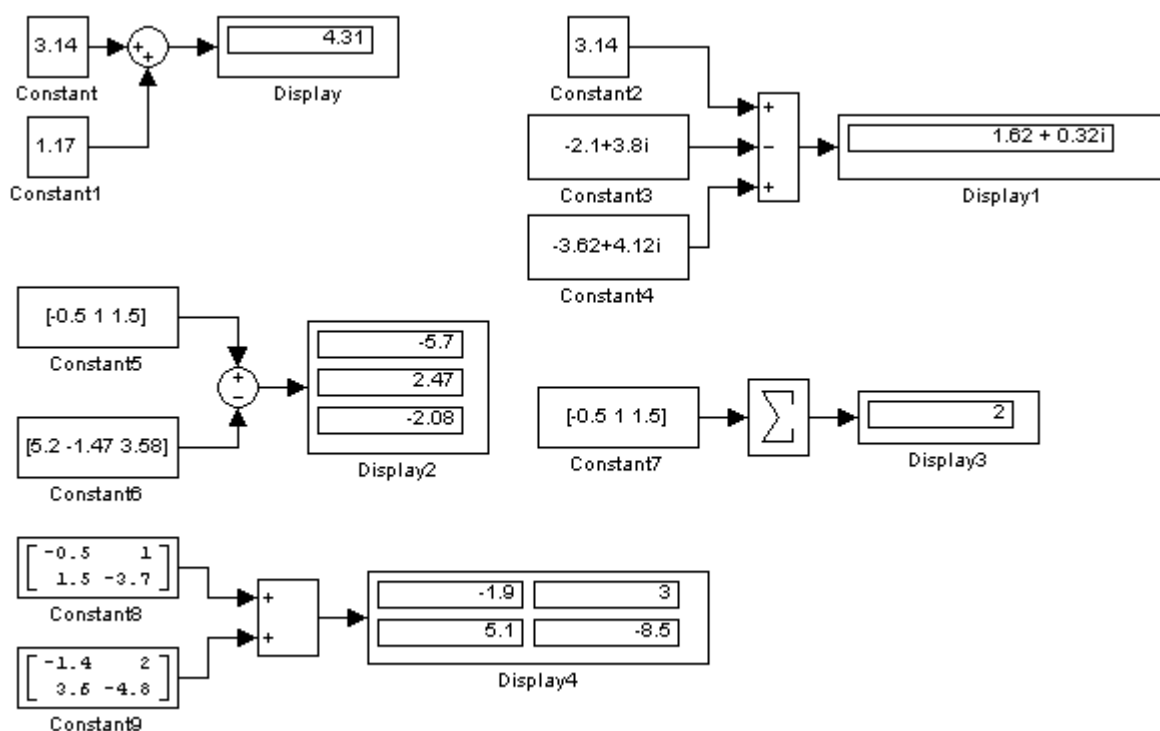


Рис. 2.2.3. Приклади використання блоку Sum

2.2.3. Блок множення Product

Призначення: Виконує обчислення добутку поточних значень сигналів.

Параметри:

Number of inputs - Кількість входів. Може здаватися як число або як список знаків. У списку знаків можна використовувати знаки * (помножити) та / (розділити).

Multiplication - Спосіб виконання операції. Може приймати значення (зі списку):

- Element-wise - Поелементний.

- Matrix - Матричний.

Saturate on integer overflow (прапорець) - Придушувати переповнення цілого.
При встановленому прапорці обмеження сигналів цілого типу виконується коректно.

Якщо параметр Number of inputs заданий списком, що включає крім знаків множення також знаки поділу, то мітки входів будуть позначені символами відповідних операцій.

Блок може використовуватися для операцій множення або ділення скалярних векторних або матричних сигналів. Типи вхідних сигналів блоку повинні збігатися. Якщо в якості кількості входів вказати цифру 1 (один вхід), то блок можна використовувати для визначення твори елементів вектора.

Приклади використання блоку Product при виконанні скалярних і поелементних операцій показані на 2.2.4.

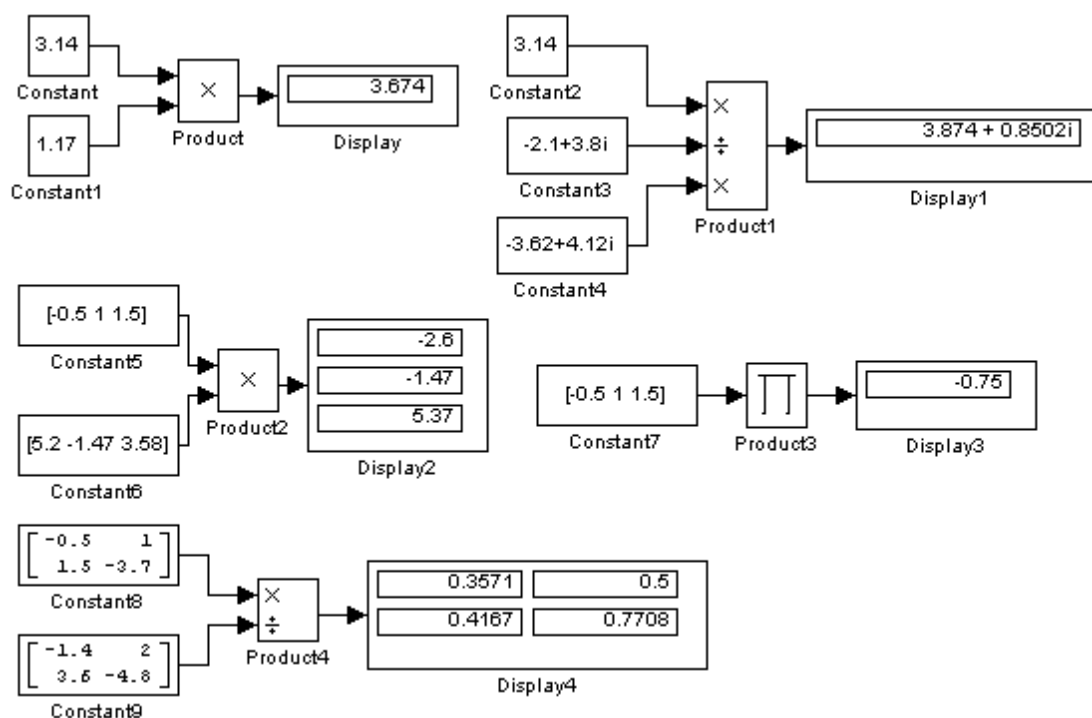


Рис. 2.2.4. Приклади використання блоку Product при виконанні скалярних і поелементних операцій

При виконанні матричних операцій необхідно дотримуватися правила їх виконання. Наприклад, при множенні двох матриць необхідно, щоб кількість рядків першої матриці дорівнювала кількості стовпців другого матриці. Приклади використання блоку Product при виконанні матричних операцій показані на рис.

2.2.5. У прикладі показані операції формування зворотної матриці, ділення матриць, а також множення матриць.

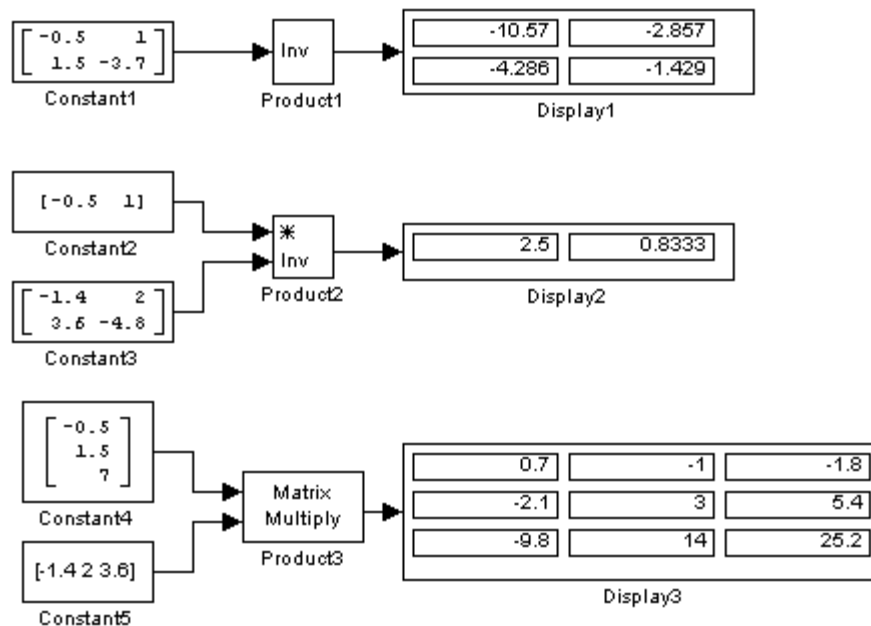


Рис. 2.2.5. Приклади використання блоку Product при виконанні матричних операцій

2.2.4. Блок визначення знака сигналу Sign

Призначення: Визначає знак вхідного сигналу.

Параметри:

Немає.

Блок працює у відповідності з наступним алгоритмом:

Якщо вхідний сигнал блоку позитивний, то вихідний сигнал дорівнює 1.

Якщо вхідний сигнал блоку негативний, то вихідний сигнал дорівнює -1.

Якщо вхідний сигнал блоку дорівнює 0, то вихідний сигнал також дорівнює 0.

Рис. 2.2.6. ілюструє роботу блоку Sign.

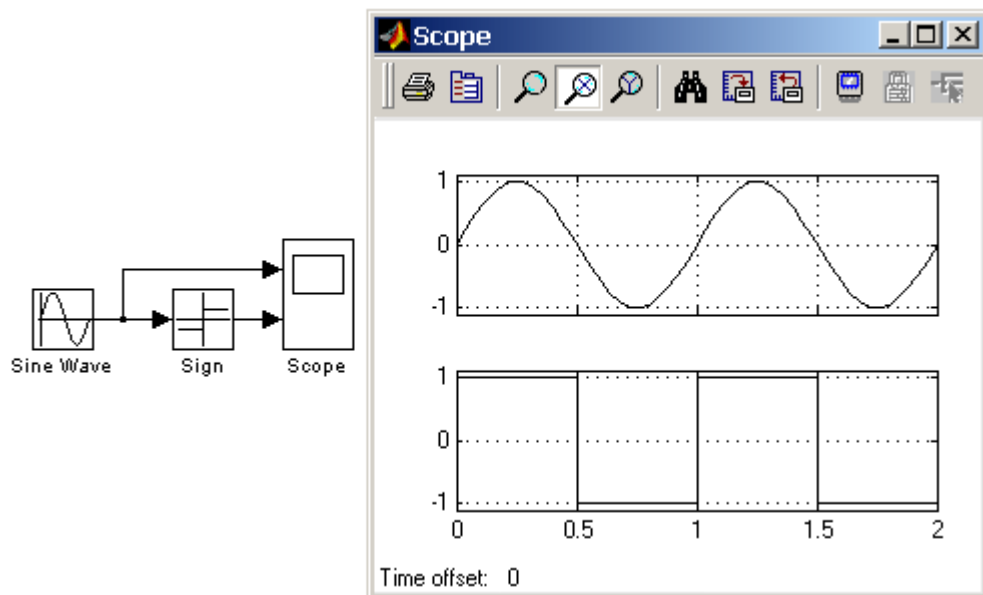


Рис. 2.2.6. Приклад використання блоку Sign

2.2.5. Підсилювачі Gain і Matrix Gain

Призначення: Виконують множення вхідного сигналу на постійний коефіцієнт.

Параметри:

Gain - Коефіцієнт посилення.

Multiplication - Спосіб виконання операції. Може приймати значення (зі списку):

- Element-wise $K * u$ - Поелементний.
- Matrix $K * u$ - Матричний. Коефіцієнт посилення є лівостороннім операндом.
- Matrix $u * K$ - Матричний. Коефіцієнт посилення є правостороннім операндом.

Saturate on integer overflow (прапорець) - Придушувати переповнення цілого. При встановленому прапорці обмеження сигналів цілого типу виконується коректно.

Блоки підсилювачів Gain і Matrix Gain є один і той же блок, але з різними початковими установками параметра Multiplication.

Параметр блоку Gain може бути позитивним чи негативним числом, як більше, так і менше 1. Коефіцієнт посилення можна задавати у вигляді скаляра, матриці або вектора, а також у вигляді обчислюваного виразу.

У тому випадку якщо параметр Multiplication заданий як Element-wise $K * u$, то блок виконує операцію множення на заданий коефіцієнт скалярного сигналу або

кожного елемента векторного сигналу. В іншому випадку блок виконує операцію матричного множення сигналу на коефіцієнт заданий матрицею.

За замовчуванням коефіцієнт посилення є дійсним числом типу double.

Для операції поелементного посилення вхідний сигнал може бути скалярним, векторним або матричним будь-якого типу, за винятком логічного (boolean). Елементи вектора повинні мати однаковий тип сигналу. Вихідний сигнал блоку буде мати той же самий тип, що і вхідний сигнал. Параметр блоку Gain може бути скаляром, вектором або матрицею будь-якого типу, за винятком логічного (boolean).

При обчисленні вихідного сигналу блок Gain використовує наступні правила:

Якщо вхідний сигнал дійсного типу, а коефіцієнт посилення комплексний, то вихідний сигнал буде комплексним.

Якщо тип вхідного сигналу відрізняється від типу коефіцієнта посилення, то Simulink намагається виконати приведення типу коефіцієнта посилення до типу вхідного сигналу. У тому випадку, якщо таке приведення неможливо, то розрахунок буде зупинений з виведенням повідомлення про помилку. Така ситуація може виникнути, наприклад, якщо вхідний сигнал є беззнакове ціле (uint8), а параметр Gain заданий негативним числом.

Приклади використання блоку Gain при виконанні скалярних і поелементних операцій показані на 2.2.7.

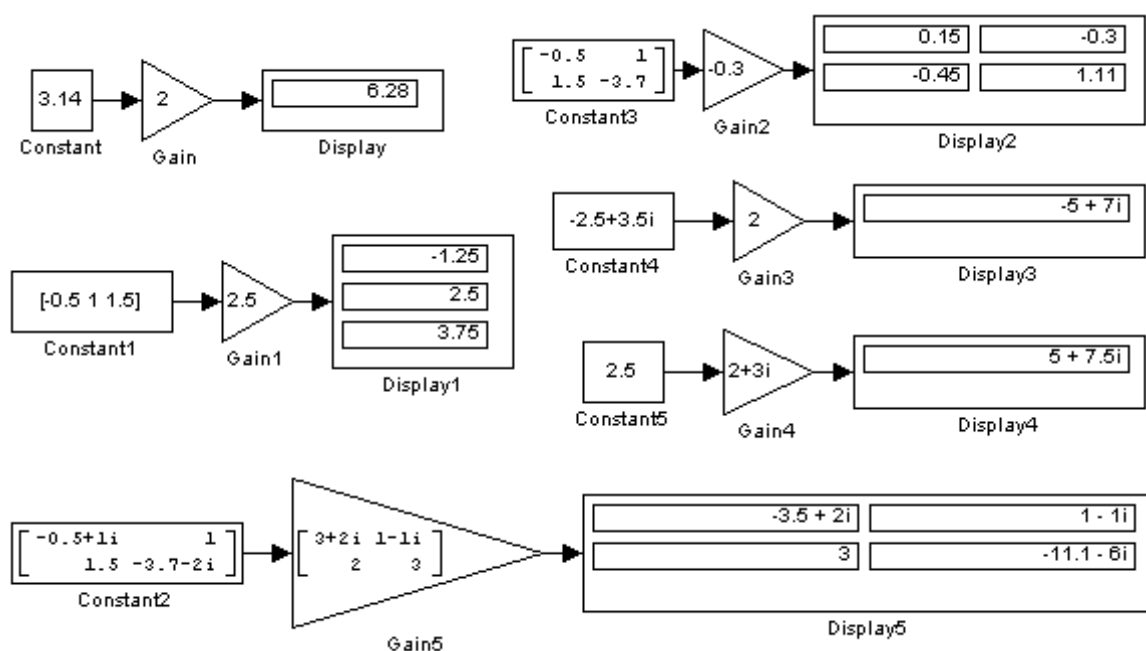


Рис. 2.2.7. Приклади використання блоку Gain.

Для операцій матричного посилення (матричного множення вхідного сигналу на заданий коефіцієнт) вхідний сигнал і коефіцієнт підсилення повинні бути скалярними, векторними або матричними значеннями комплексного чи дійсного типу single або double.

Приклади використання блоку Matrix Gain при виконанні матричних операцій показані на рис. 2.2.8.

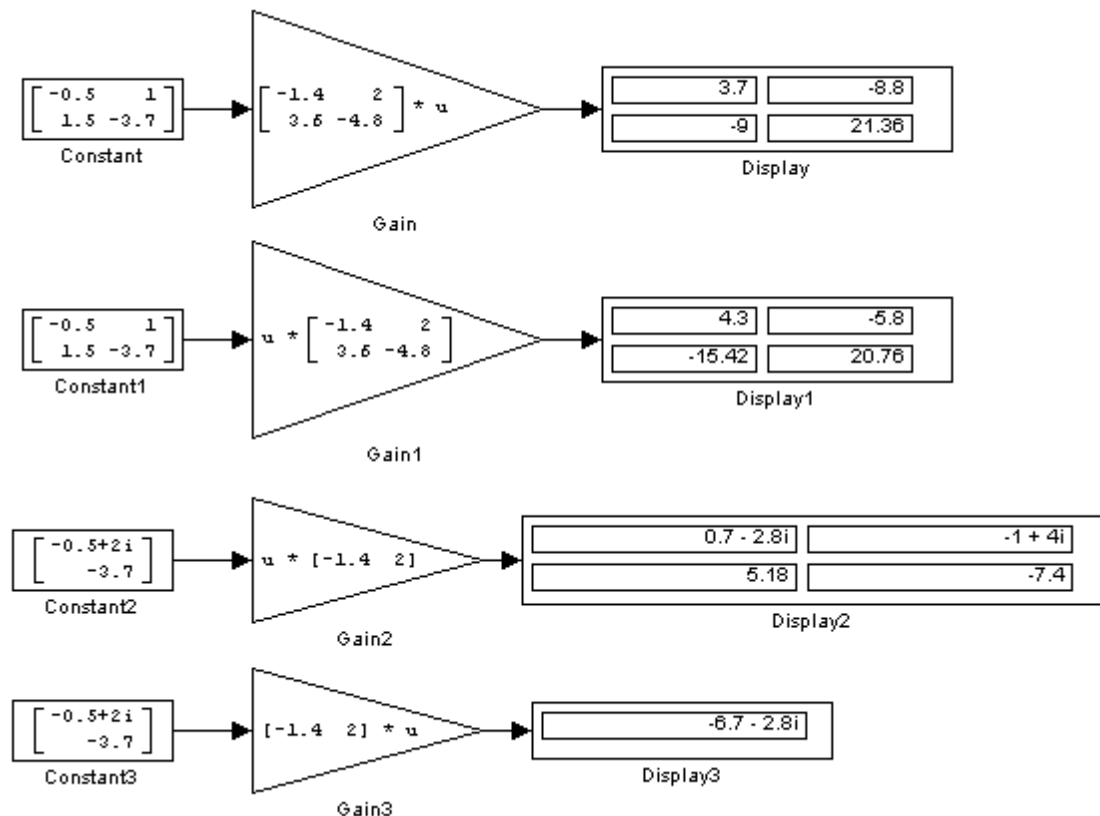


Рис. 2.2.8. Приклади використання блоку Matrix Gain

2.2.6. Повзунковий регулятор Slider Gain

Призначення: Забезпечує зміну коефіцієнта посилення в процесі розрахунку.

Параметри:

Low - Нижня межа коефіцієнта посилення.

High - Верхня межа коефіцієнта посилення.

Для зміни коефіцієнта посилення блоку Slider Gain необхідно пересунути повзунок регулятора. Переміщення повзунка вправо призведе до збільшення коефіцієнта посилення, переміщення вліво - до зменшення. Зміна коефіцієнта

посилення буде виконуватися в межах діапазону заданого параметрами Low і High.

Якщо клацнути за допомогою миші на лівій чи правій стрілках шкали регулятора, то коефіцієнт посилення зміниться на 1% від встановленого діапазону. Якщо клацнути за допомогою миші на самій шкалі регулятора ліворуч або праворуч від повзунка, то коефіцієнт посилення зміниться на 10% від встановленого діапазону. Можна також просто задати необхідне значення коефіцієнта в середньому вікні блоку.

Блок може виконувати поелементне посилення векторного або матричного сигналу. Вхідний сигнал може бути комплексним.

Приклади використання блоку Slider Gain показані на рис. 2.2.9.

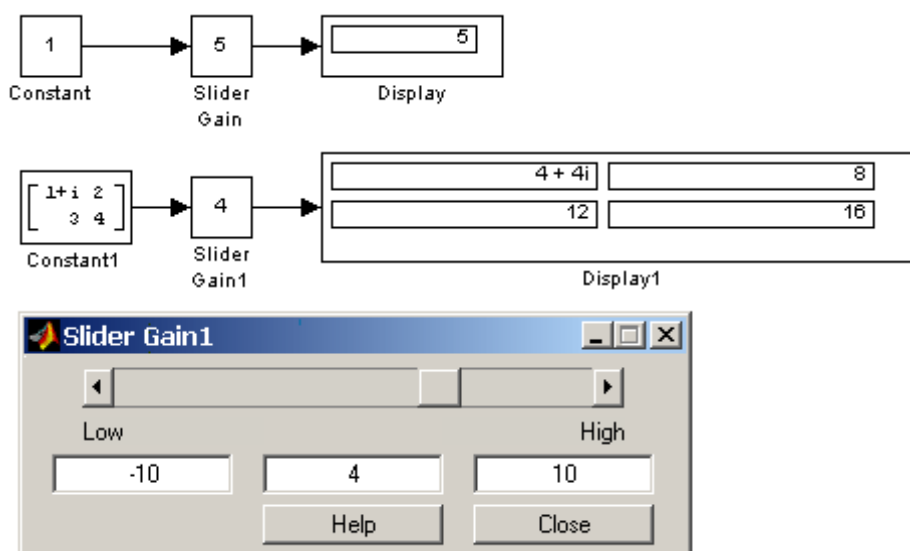


Рис. 2.2.9. Приклади використання блоку Slider Gain

2.2.7. Блок скалярного множення Dot Product

Призначення: Виконує обчислення скалярного твору (згортку) двох векторів.

Параметри:

Немає.

Блок виконує обчислення вихідного сигналу у відповідності з виразом:

$y = \text{sum}(\text{conj}(u1) .* u2)$, де $u1$ і $u2$ - вхідні вектори, conj - операція обчислення комплексно-сполученого числа, sum - операція обчислення суми.

Якщо обидва вхідних вектора є дійсними, то вихідний сигнал також буде дійсним. Якщо хоча б один із вхідних векторів містить комплексний сигнал, то вихідний сигнал буде комплексним.

Приклади, що ілюструють роботу блоку Dot Product, показані на рис. 2.2.10.

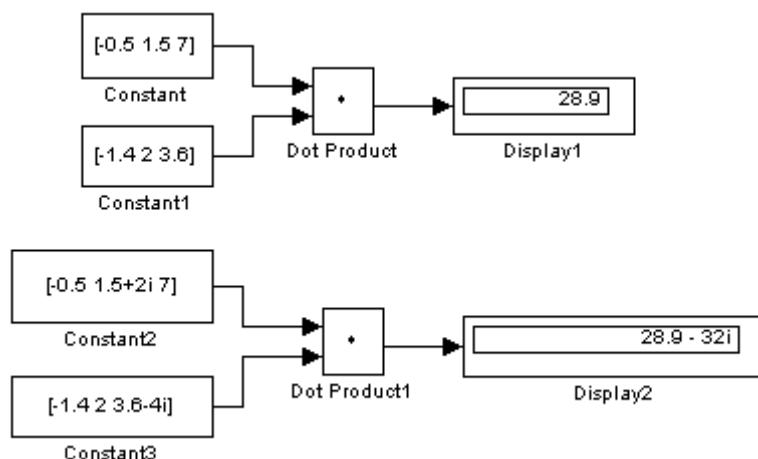


Рис. 2.2.10. Приклади використання блоку Dot Product

2.2.8. Блок обчислення математичних функцій Math Function

Призначення: Виконує обчислення математичної функції.

Параметри:

Function - Вид обчислюється функції (вибирається із списку):

exp - Експоненціальна функція

log - Функція натурального логарифма

10^u - Обчислення ступеня 10

log10 - Функції логарифма

magnitude² - Обчислення квадрата модуля вхідного сигналу

square - Обчислення квадрата вхідного сигналу

sqrt - Квадратний корінь

pow - Піднесення до степеня

conj - Обчислення комплексно-сполученого числа

reciprocal - Обчислення частки від розподілу вхідного сигналу на 1

hypot - Обчислення кореня квадратного з суми квадратів вхідних сигналів (гіпотенузи прямокутного трикутника за значеннями катетів)

rem - Функція, що обчислює залишок від ділення першого вхідного сигналу на другий

mod - Функція, що обчислює залишок від ділення з урахуванням знака

transpose - Транспонування матриці

hermitian - Обчислення ермітової матриці.

Output signal type - Тип вихідного сигналу (вибирається із списку):

auto - Автоматичне визначення типу

real-дійсний сигнал

complex- Комплексний сигнал.

Тип вихідного сигналу в залежності від типу вхідного сигналу, що обчислюється функції і параметра блоку Output signal type наведено в таблиці 2.2.1.

Таблица 2.2.1.

Функция	Входной Сигнал	Выходной Сигнал		
		Auto	Real	Complex
Exp, log, 10 ^u , log10, square, sqrt, pow, reciprocal, conjugate, transpose, hermitian	real complex	real complex	real error	complex complex
magnitude squared	real complex	real real	real real	complex complex
hypot, rem, mod	real complex	real error	real error	complex error

Приклади використання блоку Math Function показані на рис. 2.2.11.

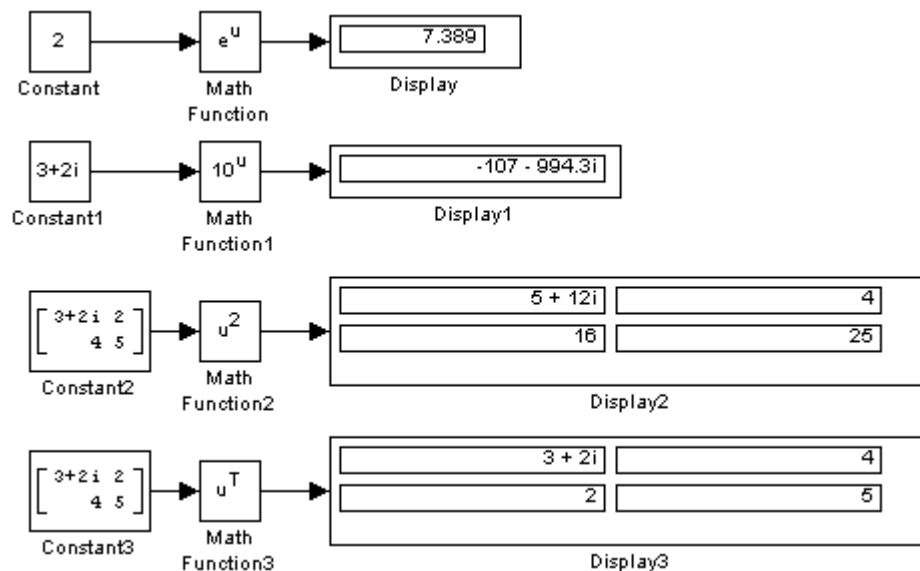


Рис. 2.2.11. Приклади використання блоку Math Function

2.2.9. Блок обчислення тригонометричних функцій Trigonometric Function

Призначення: Виконує обчислення тригонометричної функції.

Параметри:

Function - Вид обчислюється функції (вибирається із списку): sin, cos, tan, asin, acos, atan, atan2, sinh, cosh і tanh.

Output signal type - Тип вихідного сигналу (вибирається із списку):

- Auto - Автоматичне визначення типу.
- Real-дійсний сигнал.
- Complex- Комплексний сигнал.

При векторному або матричному вхідному сигналі блок виконує поелементне обчислення заданої функції.

Приклади використання блоку Trigonometric Function показані на рис. 2.2.12.

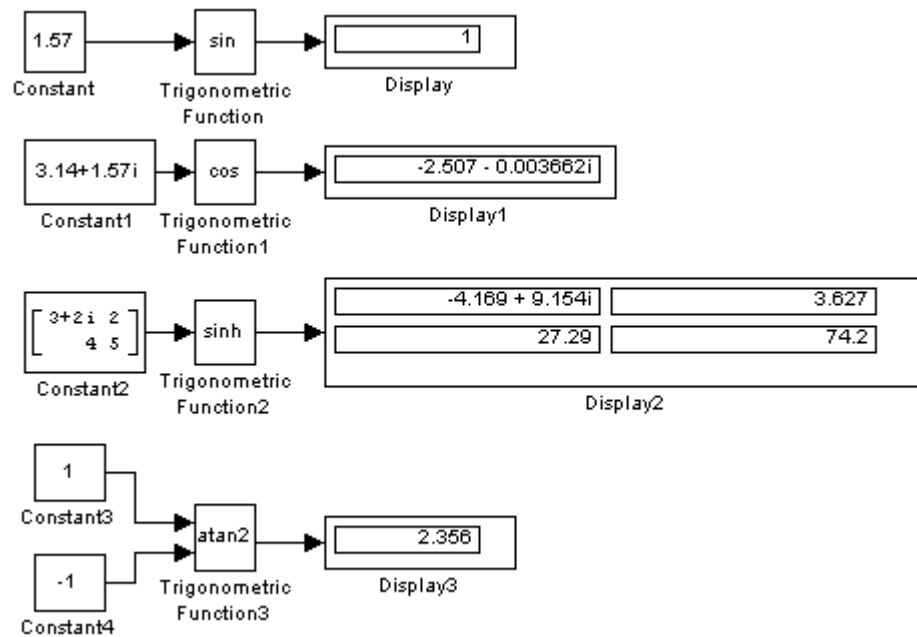


Рис. 2.2.12. Приклади використання блоку Trigonometric Function

2.2.10. Блок обчислення дійсної та (або) уявної частини комплексного числа Complex to Real-Imag

Призначення: Обчислює дійсну і (або) уявну частину комплексного числа.

Параметри:

Output - Вихідний сигнал (вибирається із списку):

Real - Дійсна частина

Image - Уявна частина

RealAndImage - Дійсна і уявна частина.

Вхідний сигнал блоку може бути скалярним, векторним або матричним сигналом.

Приклади використання блоку Complex to Real-Imag показані на рис. 2.2.13.

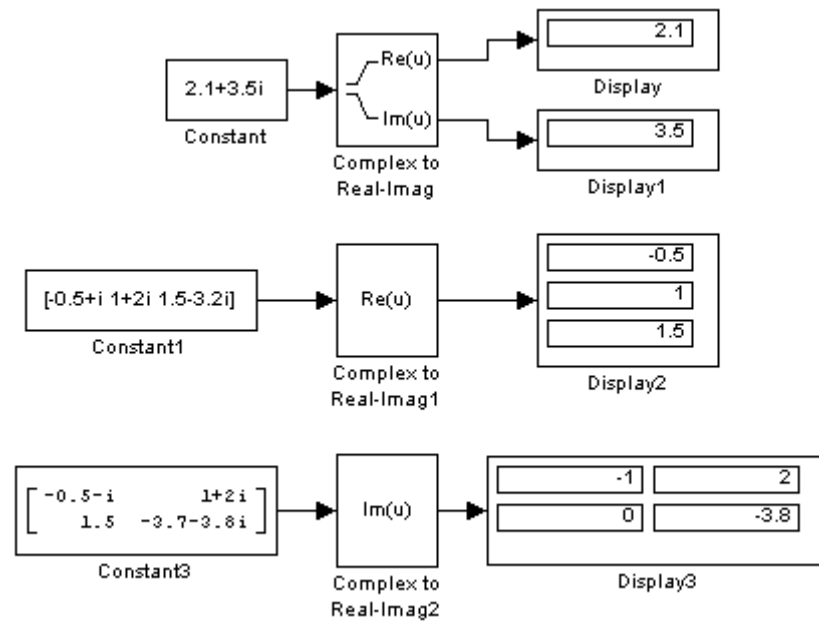


Рис. 2.2.13. Приклади використання блоку Complex to Real-Imag

2.2.11. Блок обчислення модуля і (або) аргумент комплексного числа Complex to Magnitude-Angle

Призначення: Обчислює модуль і (або) аргумент комплексного числа.

Параметри:

Output - Вихідний сигнал (вибирається із списку):

Magnitude - Модуль.

Angle - Аргумент.

MagnitudeAndAngle - Модуль і аргумент.

Вхідний сигнал блоку може бути скалярним, векторним або матричним сигналом.

Приклади використання блоку Complex to Magnitude-Angle показані на рис. 2.2.14.

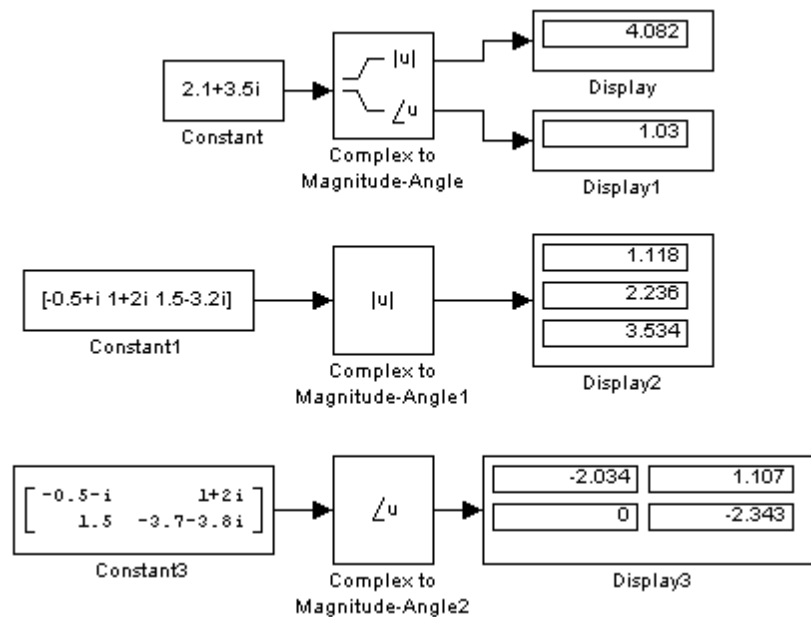


Рис. 2.2.14. Приклади використання блоку Complex to Magnitude-Angle

2.2.12. Блок обчислення комплексного числа за його дійсної та уявної частини Real-Imag to Complex

Призначення: Обчислює комплексне число за його дійсної та уявної частини.

Параметри:

Input - Вхідний сигнал (вибирається із списку):

Real - Дійсна частина.

Image - Уявна частина.

RealAndImage - Дійсна і уявна частина.

Image part - Уявна частина. Параметр доступний, якщо параметр Input оголошений як Real.

Real part - Дійсна частина. Параметр доступний, якщо параметр Input оголошений як Image.

Вхідні сигнали блоку можуть бути скалярними, векторними або матричними. Параметри Image part і Real part повинні задаватися як вектори або матриці, якщо вхідний сигнал є вектором або матрицею.

Приклади використання блоку Real-Imag to Complex показані на рис. 2.2.15.

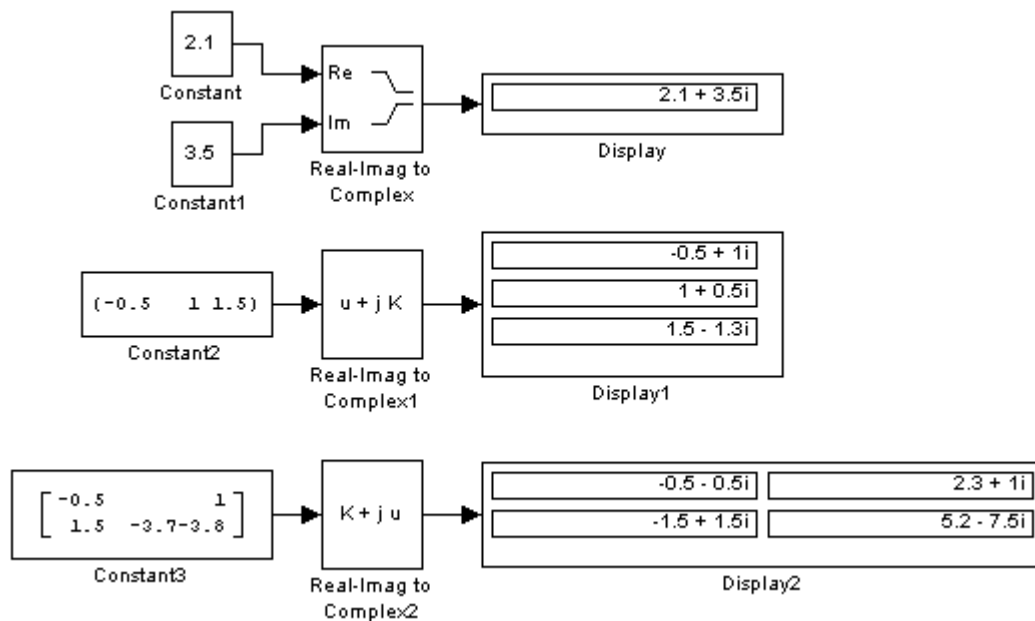


Рис. 2.2.15. Приклади використання блоку Real-Imag to Complex

2.2.13. Блок обчислення комплексного числа за його модулю і аргументу Magnitude-Angle to Complex

Призначення: Обчислює комплексне число за його модулю і аргументу.

Параметри:

Input - Вхідний сигнал (вибирається із списку):

- Magnitude - Модуль.
- Angle - Аргумент.
- MagnitudeAndAngle - Модуль і аргумент.

Angle - Аргумент. Параметр доступний, якщо параметр Input оголошений як Magnitude.

Magnitude - Модуль. Параметр доступний, якщо параметр Input оголошений як Angle.

Вхідні сигнали блоку можуть бути скалярними, векторними або матричними.

Параметри Angle і Magnitude повинні задаватися як вектори або матриці, якщо вхідний сигнал є вектором або матрицею.

Приклади використання блоку Magnitude-Angle to Complex показані на рис. 2.2.16.

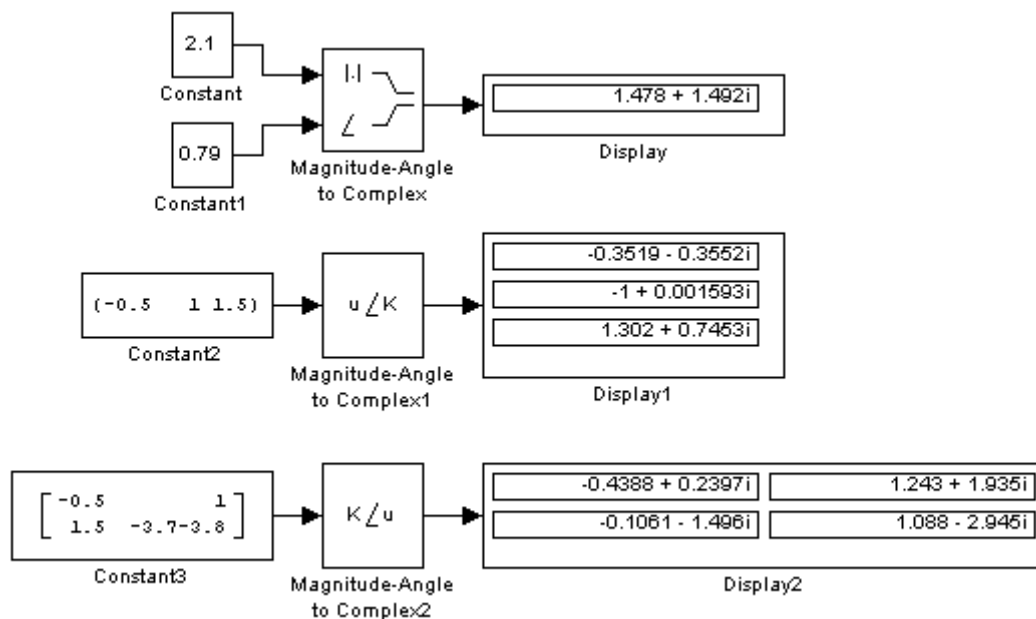


Рис. 2.2.16. Приклади використання блоку Magnitude-Angle to Complex

2.2.14. Блок визначення мінімального або максимального значення MinMax

Призначення: Визначає максимальне або мінімальне значення з усіх сигналів, що надходять на його входи.

Параметри:

Function - Вихідний параметр. Вибирається зі списку:

- Min - Мінімальне значення.
- Max - Максимальне значення.

Number of input ports - Кількість вхідних портів.

Вхідні сигнали блоку можуть бути скалярними або векторними. Блок визначає максимальне або мінімальне значення з усіх скалярних сигналів, що надходять на його входи. Якщо вхідні сигнали є векторними, то блок виконує поелементну операцію пошуку мінімального або максимального значення. У цьому випадку розмірності векторів повинні збігатися. Якщо кількість вхідних портів блоку задано рівним 1, то блок може використовуватися для знаходження мінімального або максимального значення у вхідному векторі.

Приклади використання блоку MinMax показані на рис. 2.2.17.

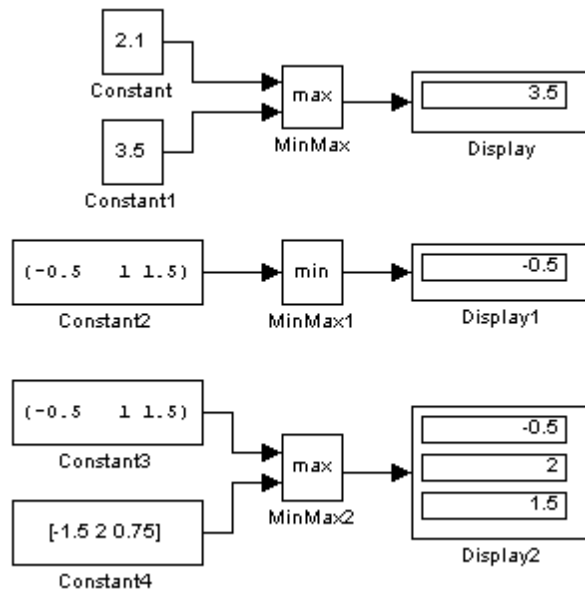


Рис. 2.2.17. Приклади використання блоку MinMax

2.2.15. Блок округлення числового значення Rounding Function

Призначення: Виконує операцію округлення числового значення.

Параметри:

Function - Спосіб округлення (вибирається із списку):

floor - Округлення до найближчого меншого цілого.

ceil - Округлення до найближчого більшого цілого.

round - Округлення до найближчого цілого.

fix - Округлення відкиданням дробової частини.

Вхідні сигнали блоку можуть бути скалярними, векторними або матричними дійсного і комплексного типу. При векторному або матричному вхідному сигналі блок виконує поелементні операції.

Вихідний сигнал блоку буде мати тип double або single.

Приклади використання блоку Rounding Function показані на рис. 2.2.18.

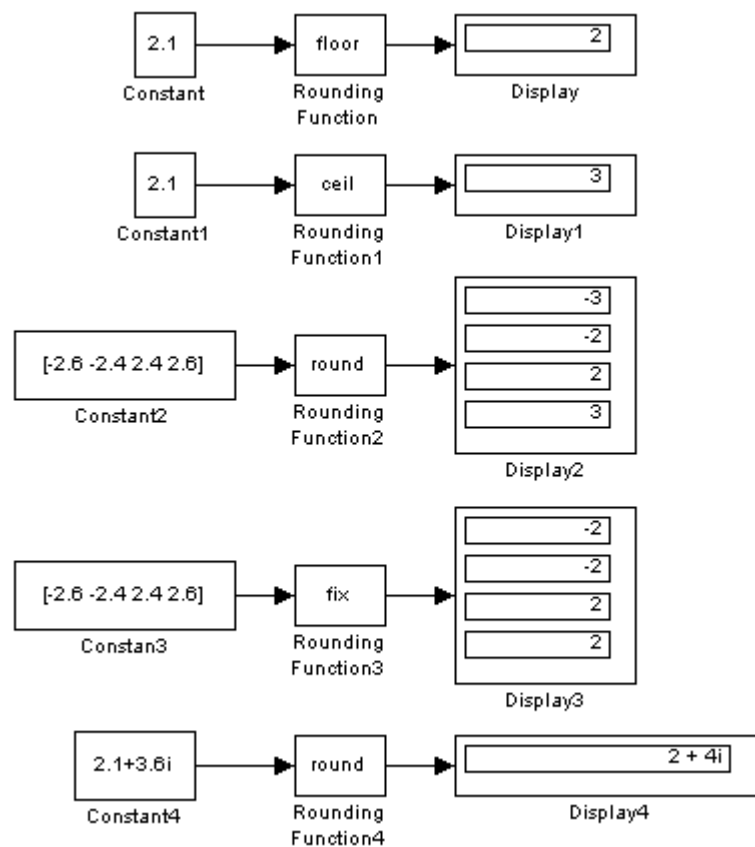


Рис. 2.2.18. Приклади використання блоку Rounding Function

Лекція 3. Знаходження коренів рівняння за допомогою Simulink. Вирішення систем алгебраїчних рівнянь за допомогою Simulink

Для знаходження коренів рівняння та вирішення систем алгебраїчних рівнянь крім вже вивчених блоків знадобляться блоки Algebraic Constraint (розділ бібліотеки Simulink: Sources), Mux (розділ бібліотеки Simulink: Signal Routing) або Vector Concatenate (розділ бібліотеки Simulink: Sources).

3.1. Блок алгебраїчного контуру Algebraic Constraint

Призначення: Виконує пошук коренів алгебраїчних рівнянь.

Параметри: Initial guess - Початкове значення вихідного сигналу.

Блок знаходить таке значення вихідного сигналу, при якому значення вхідного сигналу стає рівним нулю. При цьому вхідний сигнал повинен бути прямо або опосередковано пов'язаний з вхідним сигналом.

На рис. 3.1 показаний приклад знаходження коренів рівняння виду:

$$2 \cdot x + x^2 \cdot \exp(\log(x)) = 7$$

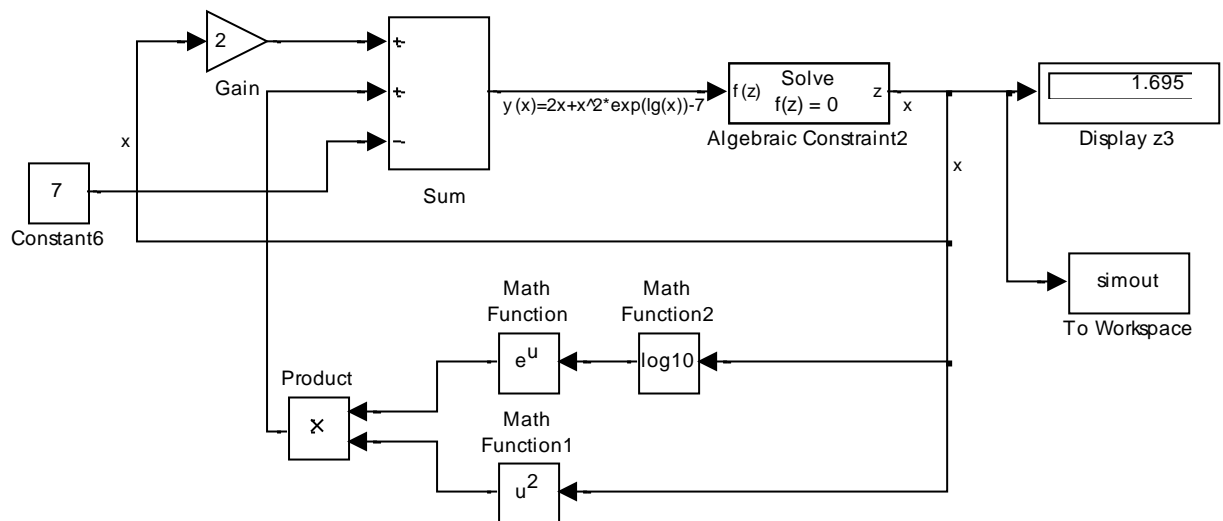


Рис. 3.1. Приклад використання блоку Algebraic Constraint для знаходження коренів рівняння

На рис. 3.2 показаний приклад вирішення системи нелінійних рівнянь виду:

$$\begin{cases} x^2 + y^2 = 6 \\ x + y = 2 \end{cases}$$

Оскільки дана система рівнянь має два рішення, то початкові значення блоків Algebraic Constraint задані у вигляді векторів. Для першого (верхнього) блоку початкове значення задано вектором $[1 \ -1]$, а для другого (нижнього) блоку - вектором $[-1 \ 1]$.

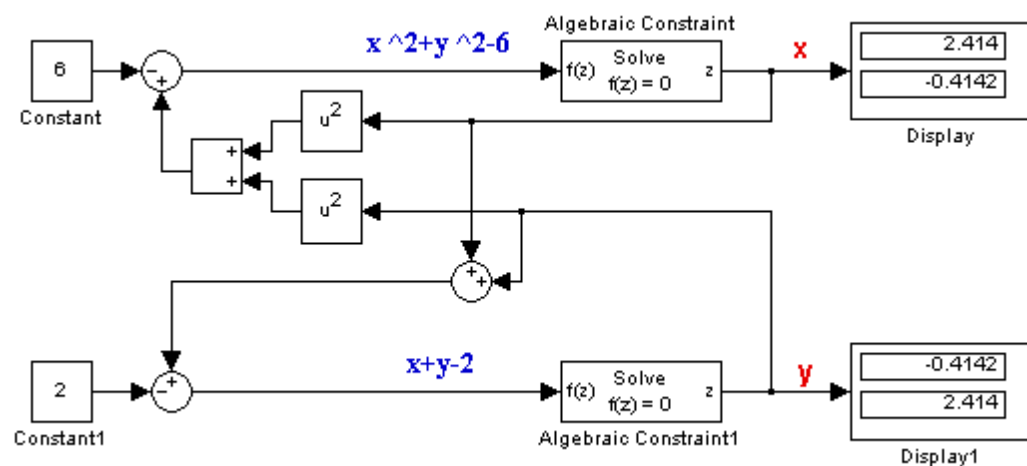


Рис. 3.2. Приклад використання блоку Algebraic Constraint для вирішення системи нелінійних рівнянь

Блок Algebraic Constraint може використовуватися також і для розв'язання нелінійних матричних рівнянь. На рис. 3.3 показаний приклад вирішення нелінійного матричного рівняння виду:

$$X^2 + 2 \cdot X + 1 = \begin{bmatrix} 13 & 4 & 4 \\ 4 & 9 & -3 \\ 4 & -3 & 57 \end{bmatrix}.$$

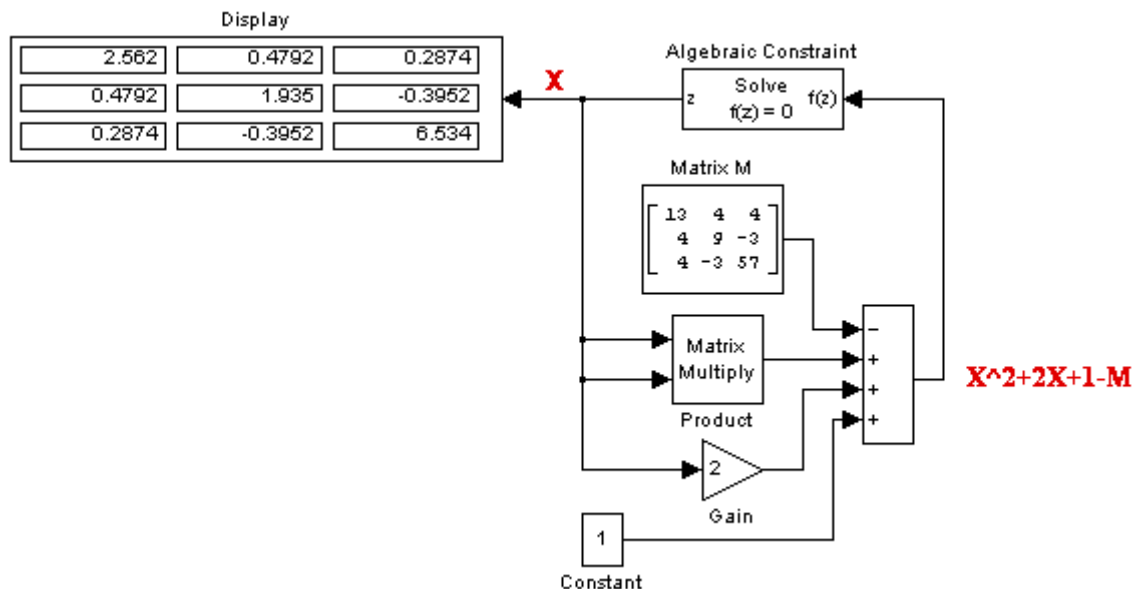


Рис. 3.3. Приклад використання блоку Algebraic Constraint для вирішення нелінійного матричного рівняння.

3.2. Блок Mux (мультиплексор, змішувач).

Призначення: Об'єднує вхідні сигнали в вектор.

Параметри:

Number of Inputs - Кількість входів.

Display option - Спосіб відображення. Вибирається зі списку:

bar - Вертикальний вузький прямокутник чорного кольору.

signals - Прямокутник з білим фоном і відображенням міток вхідних сигналів.

none - Прямокутник з білим фоном без відображення міток вхідних сигналів.

Вхідні сигнали блоку можуть бути скалярними і (або) векторними.

Якщо серед вхідних сигналів є вектори, то кількість входів можна задавати як вектор із зазначенням числа елементів кожного вектора. Наприклад, вираз $[2 \ 3 \ 1]$ визначає три вхідних сигналу, перший сигнал - вектор з двох елементів, другий сигнал - вектор з трьох елементів, і останній сигнал - скаляр. У тому випадку, якщо розмірність вхідного вектора не збігається із зазначеною в параметрі Number of Inputs, то після початку розрахунку Simulink видасть повідомлення про помилку. Розмірність вхідного вектора можна задавати як -1 (мінус один). У цьому випадку розмірність вхідного вектора може бути будь-якою.

Параметр Number of Inputs можна задавати також у вигляді списку міток сигналів, наприклад: Vector1, Vector2, Scalar. У цьому випадку мітки сигналів будуть відображатися поруч з відповідними сполучними лініями.

Сигнали, що подаються на входи блоку повинні бути одного типу (дійсного або комплексного).

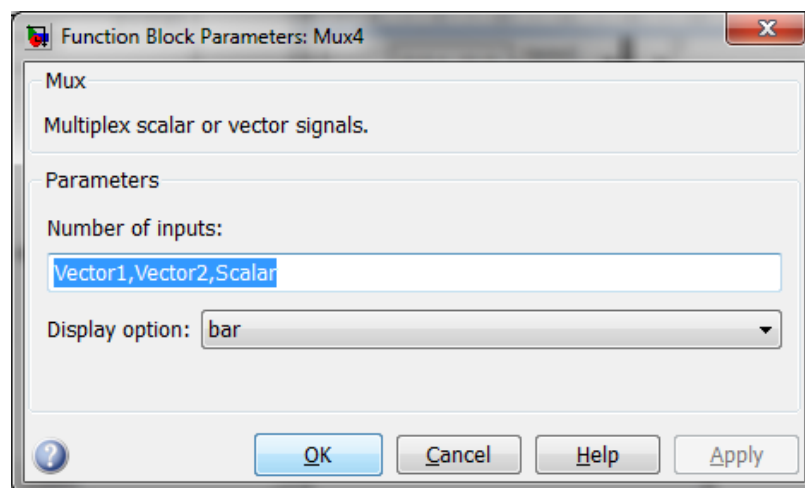
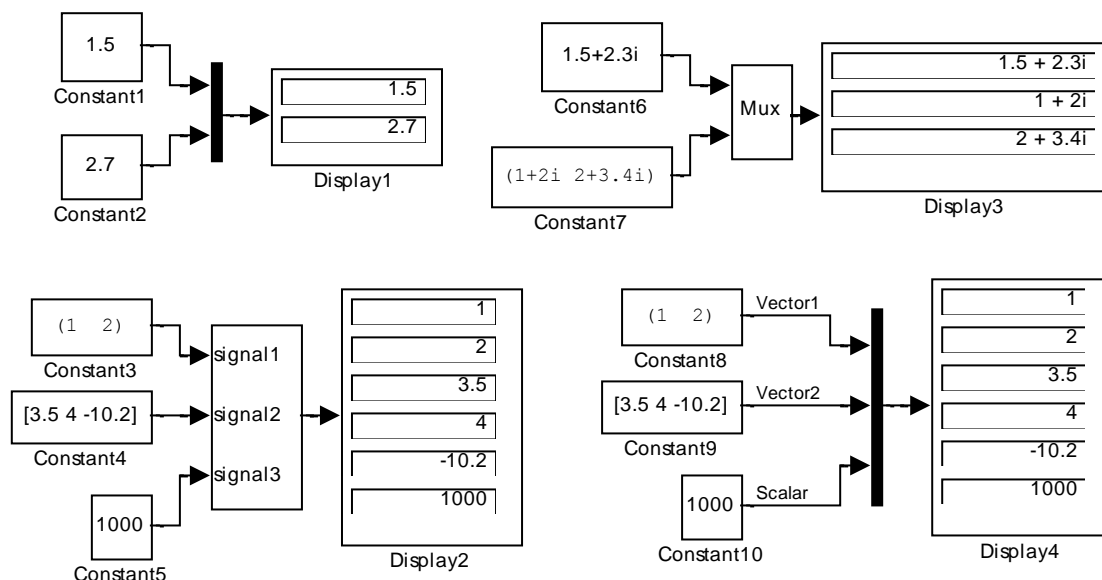


Рис. 3.3. Приклад використання блоку Mux

3.3. Блок Vector Concatenate (розділ бібліотеки Simulink: Sources).
З'єднує вхідні сигнали, створюючи безперервний вихідний сигнал/

Блок Concatenate об'єднує сигнали на своїх входах для створення вихідного сигналу, елементи якого знаходяться в суміжних ланках в пам'яті. Цей блок працює як об'єднувач і в вектор і в багатовимірний масив (рис. 3.4), залежно від настройки його параметрів. У кожному разі, блок об'єднує входи від верху до низу, або зліва направо.

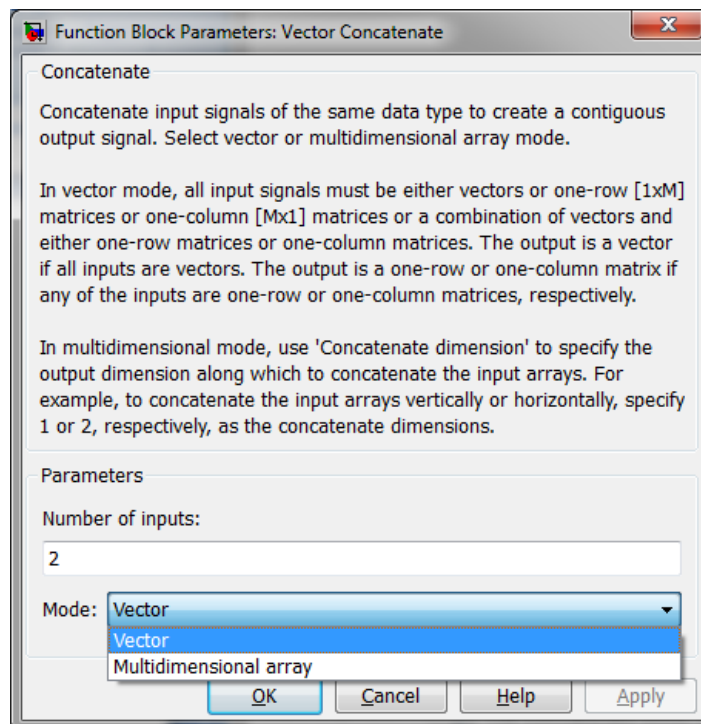


Рис. 3.4. Меню блоку Vector Concatenate

У векторному режимі, всі вхідні сигнали повинні бути або вектори або вектори-рядки $[1 \times M]$ матриці або вектори-стовпці $[M \times 1]$ матриці або комбінацію векторів і або векторів-рядків або векторів-стовпців.

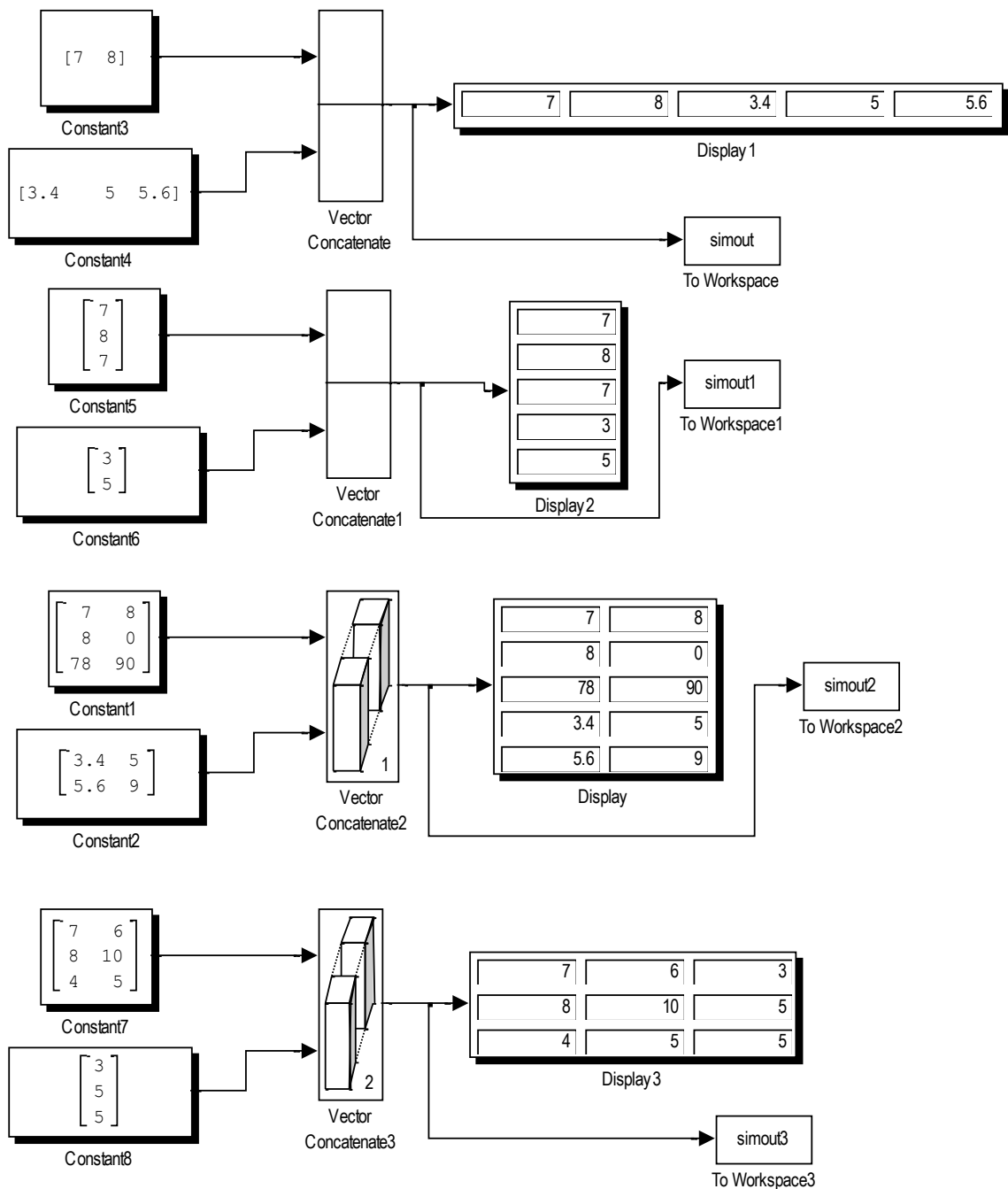


Рис. 3.3. Приклад використання блоку Vector Concatenate

Для того, щоб в змінних simout значення векторів повторювались тільки один раз (а не для кожного значення Simulation time), треба встановити Stop time рівним «0». Тоді розрахунок буде проводитись тільки для одного відліку часу.

3.4. Subsystem - підсистема.

Підсистема це фрагмент Simulink-моделі, оформлений у вигляді окремого блоку. Використання підсистем при складанні моделі має наступні позитивні сторони:

Зменшує кількість одночасно відображаються блоків на екрані, що полегшує сприйняття моделі (в ідеалі модель повністю повинна відображатися на екрані монітора).

Дозволяє створювати і налагоджувати фрагменти моделі окремо, що підвищує технологічність створення моделі.

Дозволяє створювати власні бібліотеки.

Дає можливість синхронізації паралельно працюючих підсистем.

Дозволяє включати в модель власні довідкові засоби.

Дає можливість пов'язувати підсистему з яким-небудь m-файлом, забезпечуючи запуск цього файлу при відкритті підсистеми (нестандартне відкриття підсистеми).

Використання підсистем та механізму їх блоків дозволяє створювати блоки, які не поступаються стандартним за своїм оформленням (власне вікно параметрів блоку, піктограма, довідка тощо).

Кількість підсистем в моделі не обмежена, крім того підсистеми можуть включати в себе інші підсистеми. Рівень вкладеності підсистем один в одного також не обмежений.

Зв'язок підсистеми з моделлю (або підсистемою верхнього рівня ієрархії) виконується за допомогою входних (блок Inport бібліотеки Sources) і вихідних (блок Outport бібліотеки Sinks) портів. Додавання в підсистему входного або вихідного порту призводить до появи на зображенні підсистеми мітки порту, за допомогою якої зовнішні сигнали передаються всередину підсистеми або виводяться в основну модель. Переименування блоків Inport або Outport дозволяє змінити мітки портів, які відображаються на піктограмі підсистеми зі стандартних (In і Out) на ті, які потрібні користувачеві.

Підсистеми можуть бути віртуальними (Subsystem) і монолітними (Atomic Subsystem). Відмінність цих видів підсистем полягає в порядку виконання блоків під час розрахунку. Якщо підсистема є віртуальною, то Simulink ігнорує наявність кордонів, які відокремлюють таку підсистему від моделі при визначенні порядку розрахунку блоків. Іншими словами у віртуальній системі спочатку можуть бути розраховані вихідні сигнали декількох блоків, потім виконаний розрахунок блоків в

основний моделі, а потім знову виконаний розрахунок блоків, які входять в підсистему. Монолітна підсистема вважається єдиним (неподільним) блоком і Simulink виконує розрахунок всіх блоків в такій підсистемі, не перемикаючись на розрахунки інших блоків в основний моделі. Зображення монолітної підсистеми має більш товсту рамку в порівнянні з віртуальною підсистемою.

Підсистеми можуть бути також керованими або некерованими. Керовані підсистеми завжди є монолітними. Керовані підсистеми мають додаткові (керуючі) входи, на які надходять сигнали активізації даної підсистеми. Керуючі входи розташовані зверху чи знизу підсистеми. Коли керована підсистема активізована - вона виконує обчислення. У тому випадку якщо керована підсистема пасивна, то вона не виконує обчислення, а значення сигналів на її виходах визначаються налаштуваннями вихідних портів.

Для створення в моделі підсистеми можна скористатися двома способами:

Скопіювати потрібну підсистему з бібліотеки Subsystem в модель.

Виділити за допомогою миші потрібний фрагмент моделі і виконати команду Create Subsystem з меню Edit вікна моделі (рис. 3.5). Виділений фрагмент буде поміщений в підсистему, а входи і виходи підсистеми будуть забезпечені відповідними портами (рис. 3.6). Даний спосіб дозволяє створити віртуальну некеровану підсистему. Надалі, якщо це необхідно, можна зробити підсистему монолітною, змінивши її параметри, або керованою, додавши керуючий елемент з потрібною підсистеми, яка знаходиться в бібліотеці. Скасувати угруповання блоків в підсистему можна командою Undo.

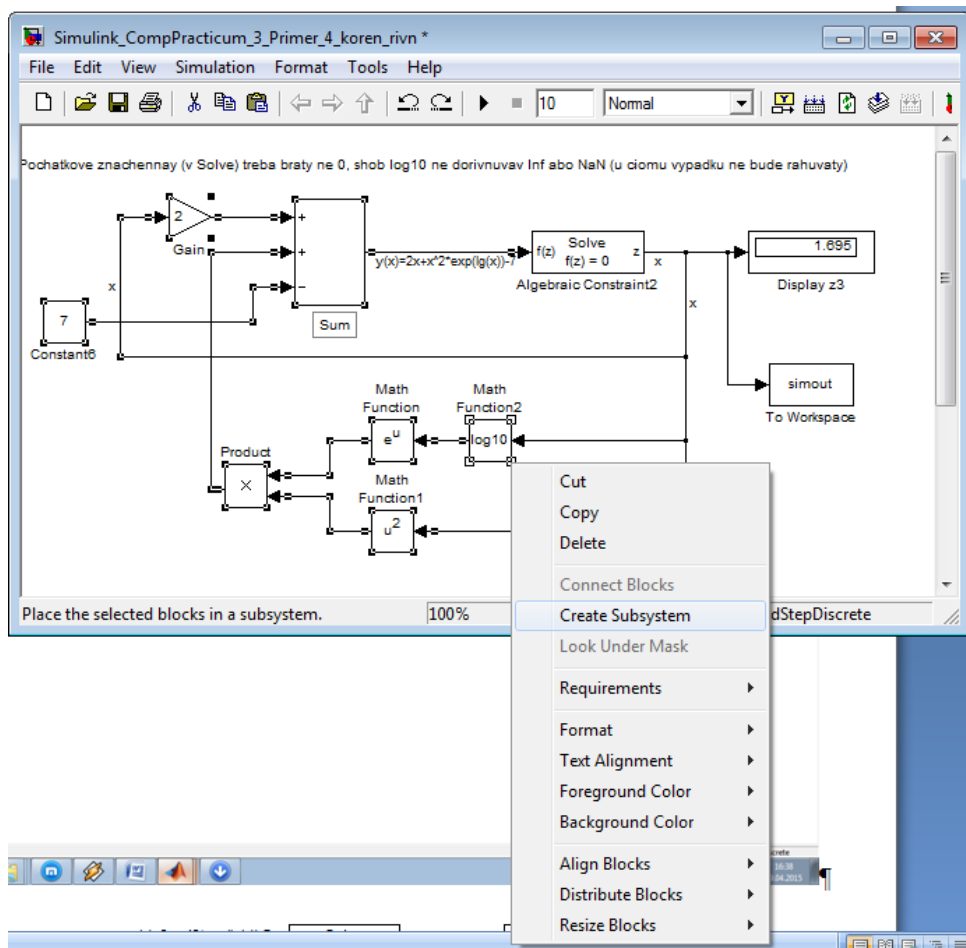
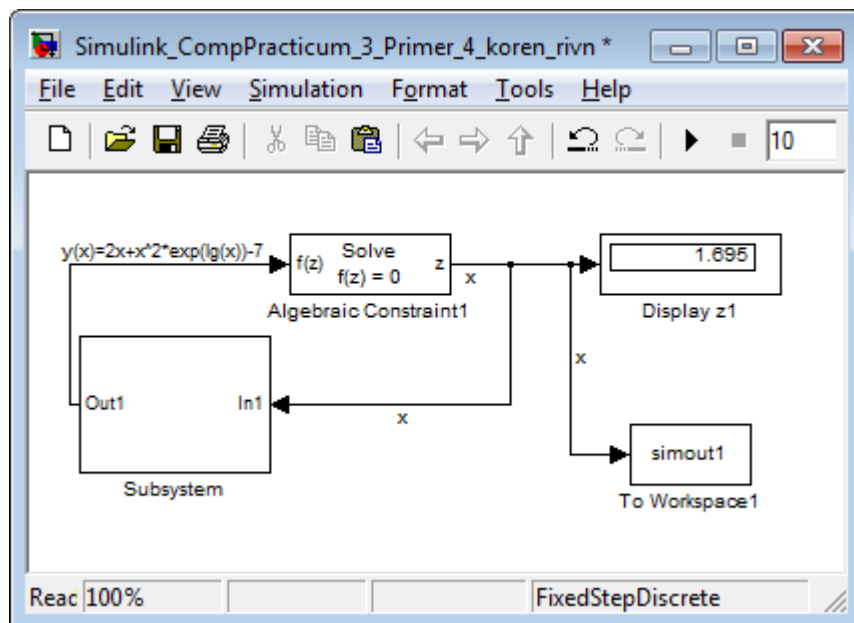
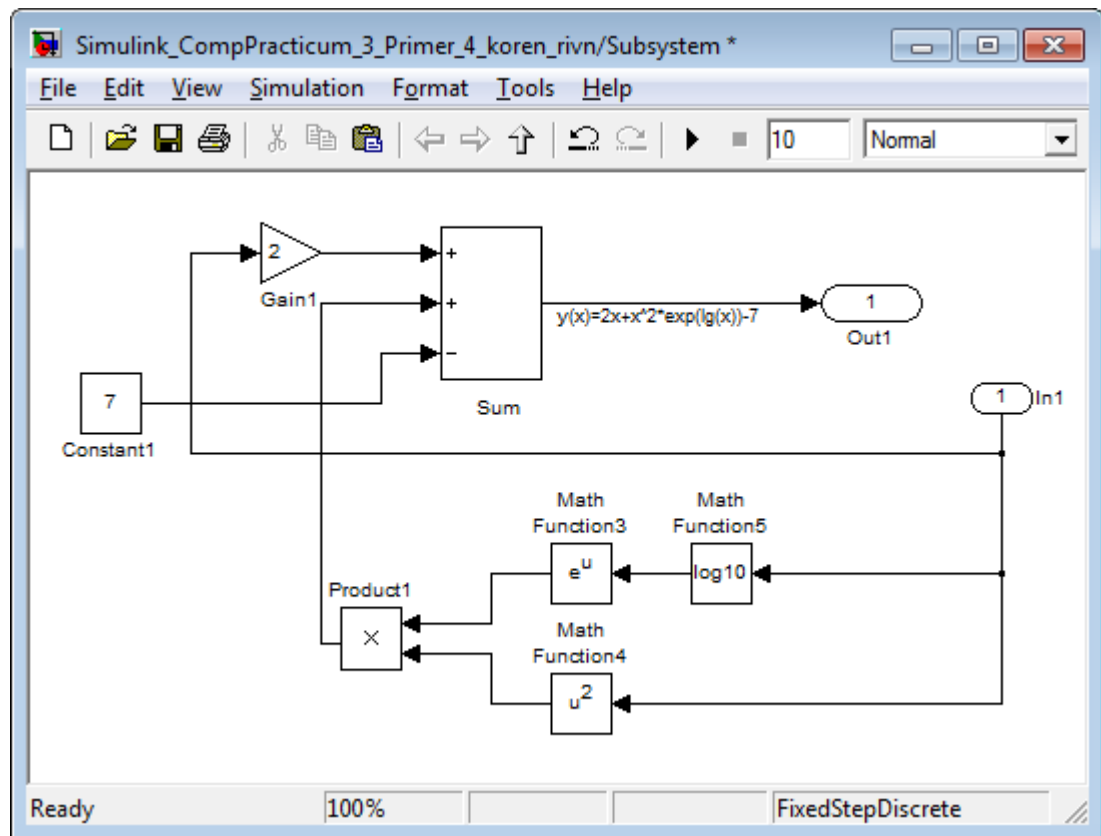


Рис. 3.5. Приклад створення підсистеми





б

Рис. 3.6. Підсистема (б) в моделі (а) для знаходження коренів рівняння $2 \cdot x + x^2 \cdot \exp(\log(x)) = 7$

Лекція 4.

Вирішення диференціальних рівнянь та систем диференціальних рівнянь за допомогою Simulink

Для знаходження коренів **диференціальних рівнянь та систем диференціальних рівнянь** крім вже вивчених блоків знадобиться блок Integrator (розділ бібліотеки Simulink: Continuous).

4.1. Інтегруючий блок Integrator

Призначення:

Виконує інтегрування вхідного сигналу.

Параметри:

- External reset - Зовнішній скидання. Тип зовнішнього керуючого сигналу, що забезпечує скидання інтегратора до початкового стану. Вибирається зі списку:

1. none - ні (скидання не виконується),
2. rising - наростаючий сигнал (передній фронт сигналу),
3. falling - спадаючий сигнал (задній фронт сигналу),
4. either - наростаючий або спадаючий сигнал,
5. level - не нульовий сигнал (скидання виконується якщо сигнал на керуючому вході стає не рівним нулю);

У тому випадку, якщо обраний який-небудь (але не none), тип керуючого сигналу, то на зображенні блоку з'являється додатковий керуючий вхід. Поруч з додатковим входом буде показано умовне позначення керуючого сигналу.

- Initial condition source - Джерело початкового значення вихідного сигналу. Вибирається зі списку:

1. internal - внутрішній
2. external - зовнішній. У цьому випадку на зображенні блоку з'являється додатковий вхід, позначений x_0 , на який необхідно подати сигнал, який задає початкове значення вихідного сигналу інтегратора.

- Initial condition - Початкова умова. Установка початкового значення вихідного сигналу інтегратора. Параметр доступний, якщо вибрано внутрішнє джерело початкового значення вихідного сигналу.

- Limit output (прапорець) - Використання обмеження вихідного сигналу.

- Upper saturation limit - Верхній рівень обмеження вихідного сигналу. Може бути заданий як числом, так і символьної послідовністю inf, тобто +.

- Lower saturation limit - Нижній рівень обмеження вихідного сигналу. Може бути заданий як числом, так і символьної послідовністю inf, тобто -.

- Show saturation port - керує відображенням порту, який виводить сигнал, що свідчить про вихід інтегратора на обмеження. Вихідний сигнал даного порту може приймати наступні значення:

1. Нуль, якщо інтегратор не перебуває на обмеженні.
2. +1, якщо вихідний сигнал інтегратора досяг верхньої обмежуючої межі.
3. -1, якщо вихідний сигнал інтегратора досяг нижньої обмежуючої межі.

- Show state port (прапорець) - Відобразити / приховати порт стану блоку. Даний порт використовується в тому випадку, якщо вихідний сигнал інтегратора потрібно подати в якості сигналу зворотного зв'язку цього ж інтегратора. Наприклад, при установці початкових умов через зовнішній порт або при скиданні інтегратора через порт скидання. Вихідний сигнал з цього порту може використовуватися також для організації взаємодії з керованою підсистемою.

- Absolute tolerance - Абсолютна похибка.

На рис. 4.1 показаний приклад роботи інтегратора при подачі на його вхід ступеневої сигналу. Початкова умова прийнята рівною нулю.

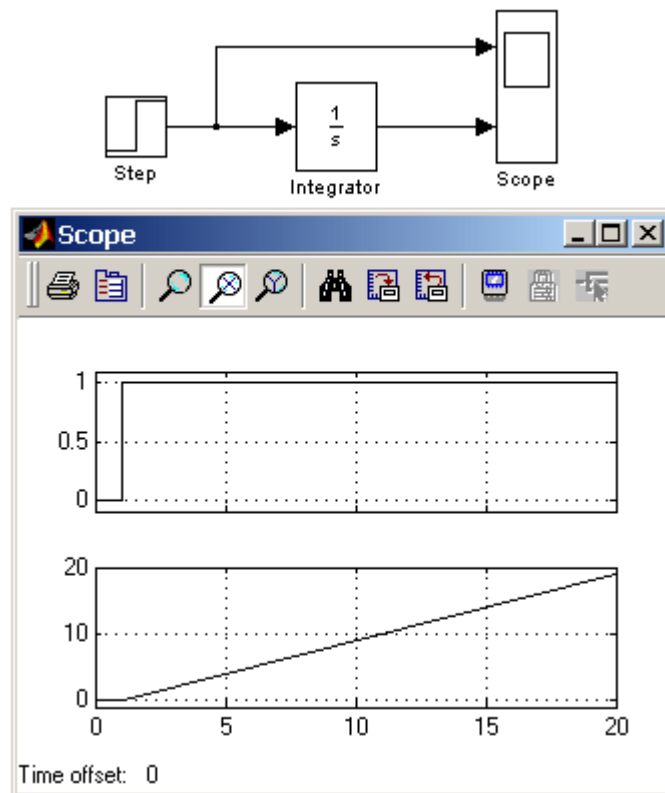


Рис. 4.1. Інтегрування ступеневого сигналу.

Приклад на рис. 4.2 відрізняється від попереднього подачею початкового значення через зовнішній порт. Початкове значення вихідного сигналу в даному прикладі задано рівним -10.

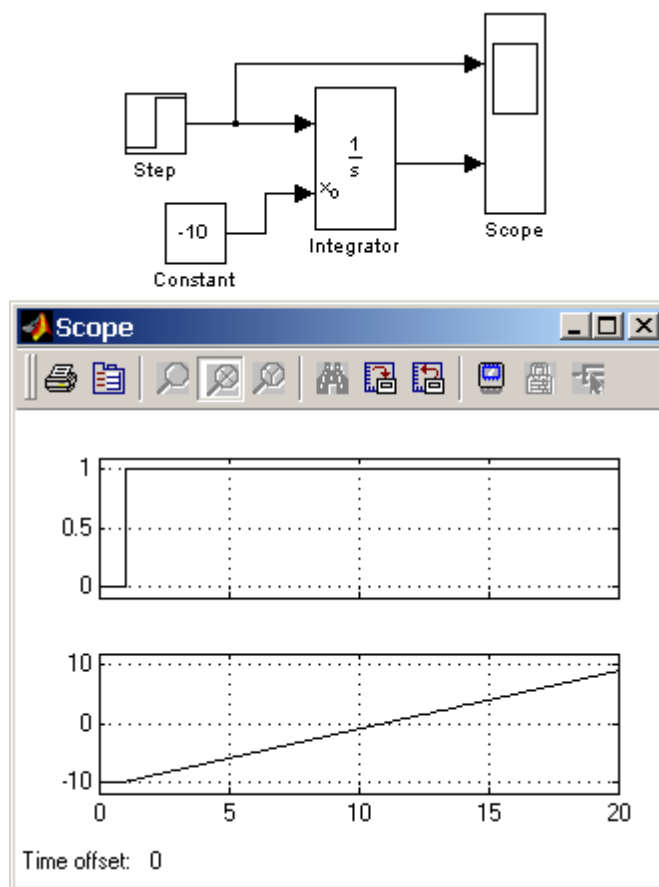


Рис. 4.2. Інтегрування ступеневого сигналу з установкою початкового значення вихідного сигналу.

Приклад на рис. 4.3 демонструє використання вхідного порту для скидання вихідного сигналу і порту стану інтегратора з метою організації зворотного зв'язку. Схема працює таким чином: вхідний постійний сигнал перетворюється інтегратором в лінійно-змінюючийся, після досягнення вихідним сигналом значення рівного 1 блок Relational Operator виробляє логічний сигнал, по передньому фронту якого відбувається скидання вихідного сигналу інтегратора до початкового значення рівного нулю. У результаті на виході інтегратора формується пилоподібний сигнал, що змінюється від 0 до +1.

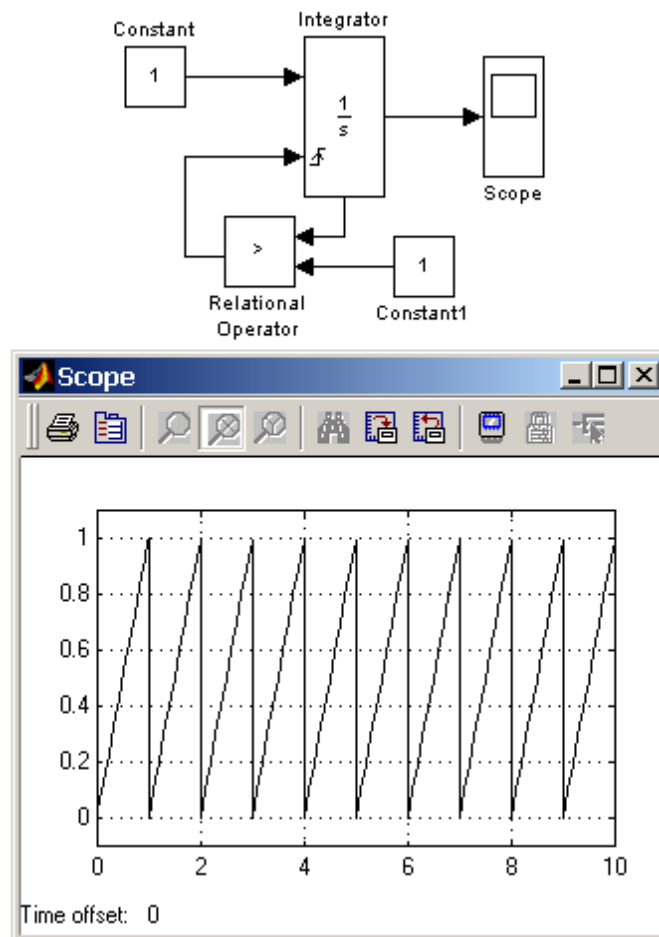


Рис. 4.3. Генератор пилкоподібної сигналу на основі інтегратора.

Наступна схема (рис. 4.4) використовує установку початкового значення інтегратора за допомогою його вихідного сигналу. У перший момент часу початкове значення вихідного сигналу інтегратора за допомогою блоку IC (Initial Condition) встановлюється рівним нулю. По досягненні вихідним сигналом значення рівного 1 блок Relational Operator подає сигнал скидання вихідного сигналу інтегратора на початковий рівень, при цьому сигналом, що задає початковий рівень, виявляється інвертований вихідний сигнал інтегратора (тобто -1). Далі цикл роботи схеми повторюється. На відміну від попередньої схеми вихідним сигналом генератора є двополярний сигнал.

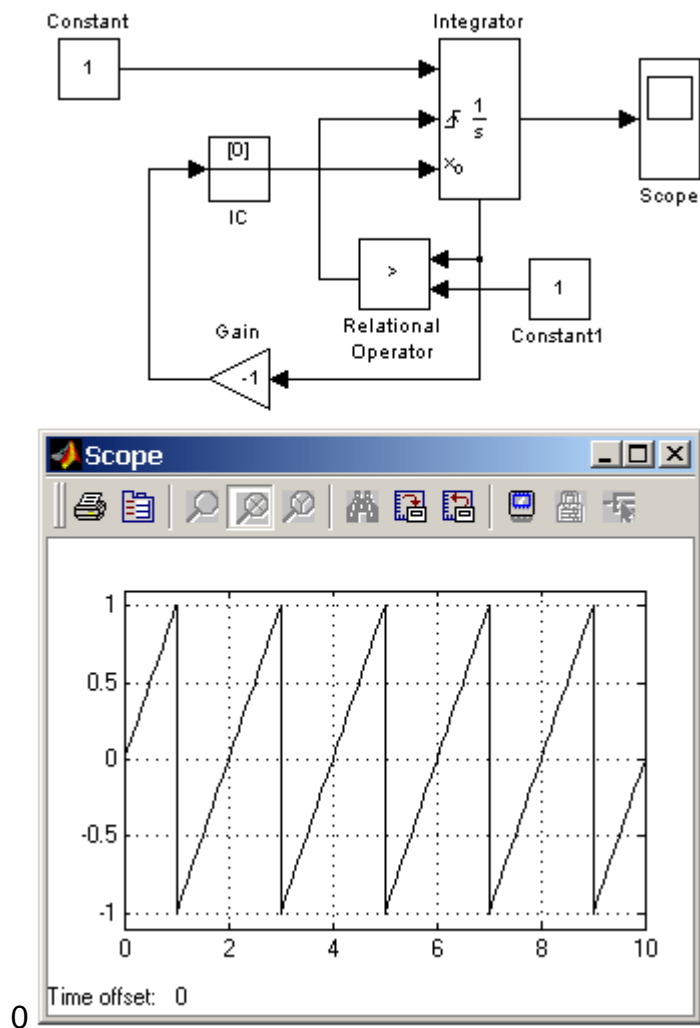


Рис. 4.4. Генератор двополярного пилоподібного сигналу на основі інтегратора.

<http://exponenta.ru/educat/systemat/semenenko/odu/index.asp>

Рішення диференціальних рівнянь (ДР) першого порядку.

Основою для вирішення звичайних диференціальних рівнянь першого порядку є задача Коші:

$$\frac{dx}{dt} = f(x, t)$$

з однією залежною змінною $x(t)$.

Приклад 1. Дано диференціальне рівняння $x'(t) + 2x(t) = \sin(t)$, $x(0) = 0$

Для побудови схеми рішення рівняння в Simulink використовується блок Integrator. На його вхід подається похідна, а на виході отримують величину x . Блоки Sum і Gain необхідні для формування значення x' у відповідності до ДР.

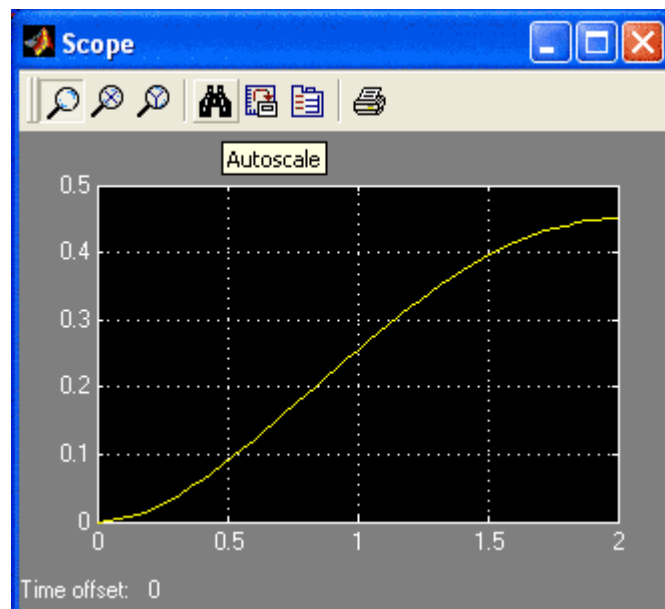
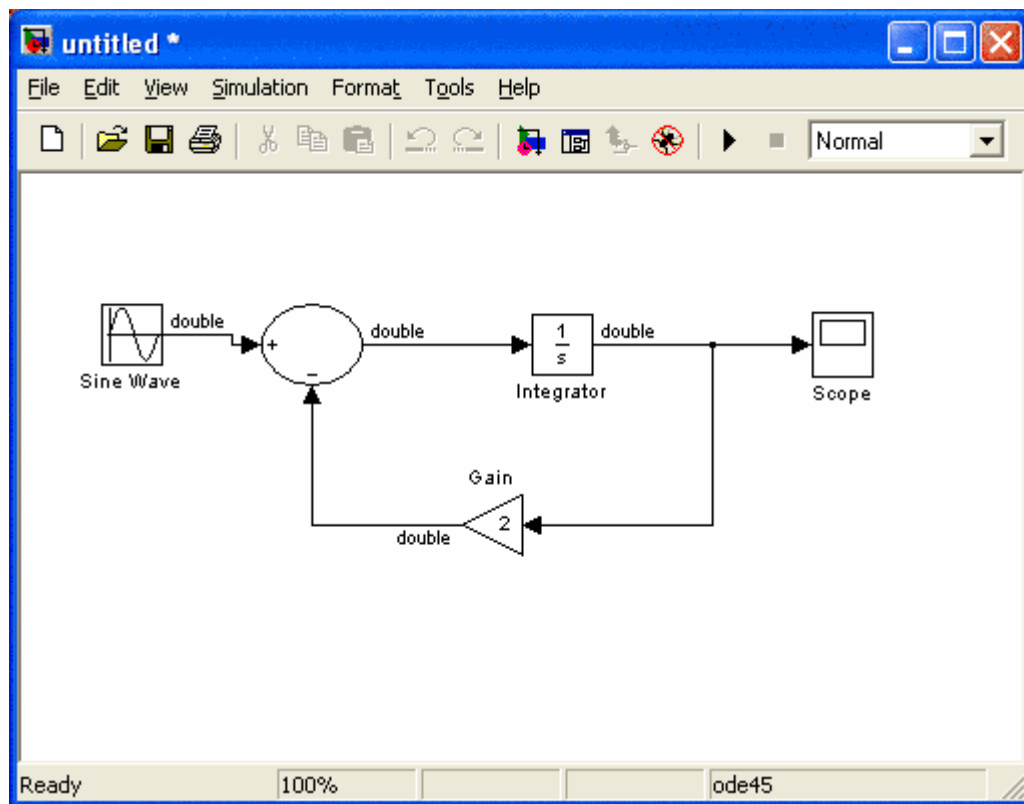


Рис. 4.5. Модель диференційного рівняння $x'(t) + 2x(t) = \sin(t)$, $x(0) = 0$ та графік його рішення

Рішення диференціальних рівнянь (ДР) другого порядку.

Приклад 2. Тепер це рівняння описує вимушений рух маятника під дією керуючого впливу $u(t)$. Керуючий сигнал можна отримати або безпосередньо в Simulink, або сформувати його в робочому просторі MATLAB і звідти передати в Simulink. Скористаємося першою можливістю. Дано диференціальне рівняння $y''(t) + \sin(y) = e^{-t}$, початкові умови: $y(0) = 1, y'(0) = 1$.

На рис. 4.6 приведена модель для вирішення цього рівняння.

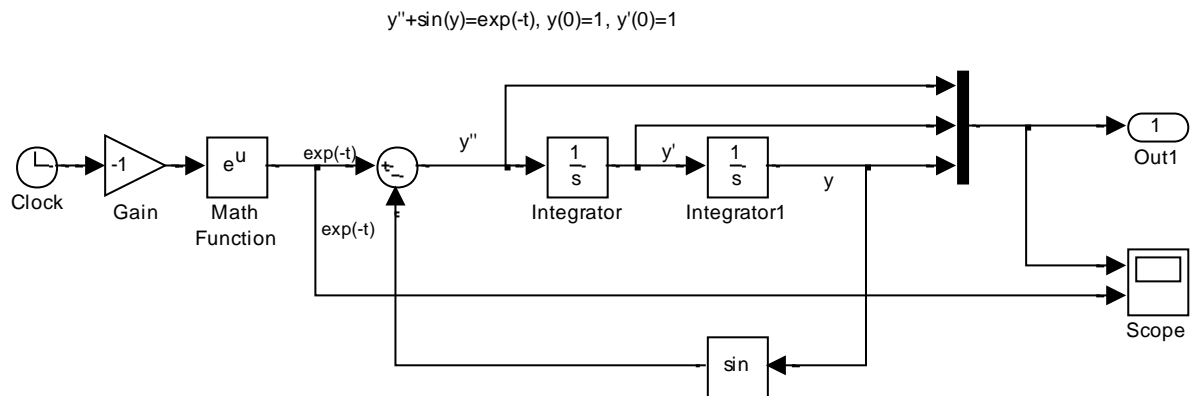


Рис. 4.6. Модель диференціального рівняння $y''(t) + \sin(y) = e^{-t}, y(0) = 1, y'(0) = 1$

Рішення диференціальних рівнянь (ДР) третього порядку

Приклад 3. Синтезувати структурну схему моделі диференціального рівняння

3 порядку: $\frac{d^3 y}{dx^3} + 0.5 \frac{d^2 y}{dx^2} + 0.9 \frac{dy}{dx} + 2y = 0$ $y(0) = 2$, 3 початковими умовами:

$y'(0) = -1, y''(0) = 0$

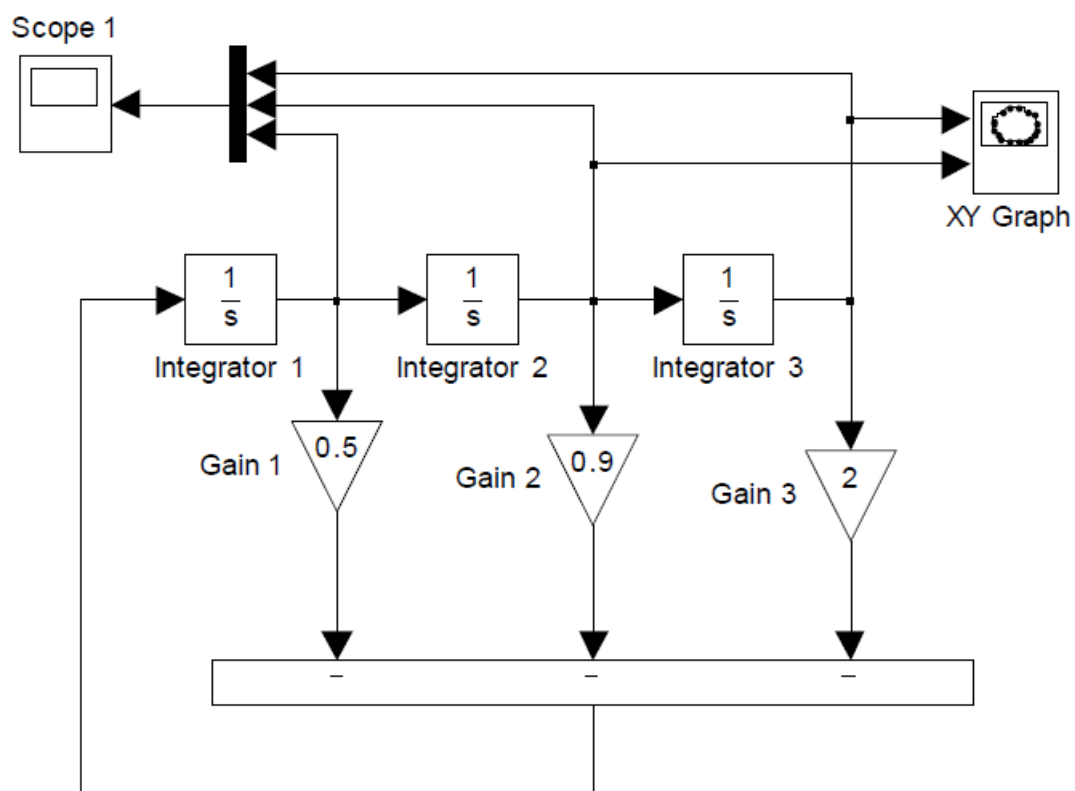


Рис. 4.7. Схема до прикладу 3

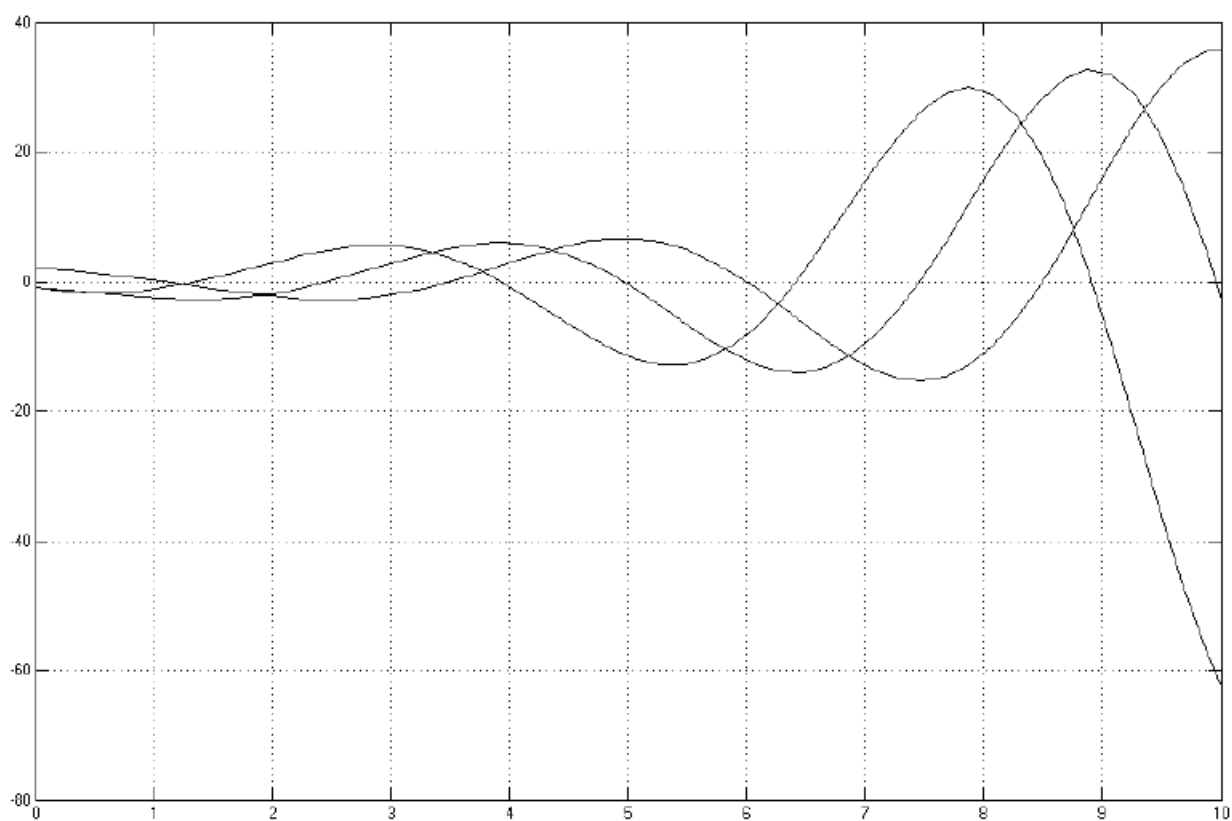


Рис. 4.7. Рішення прикладу 3 на осцилографі

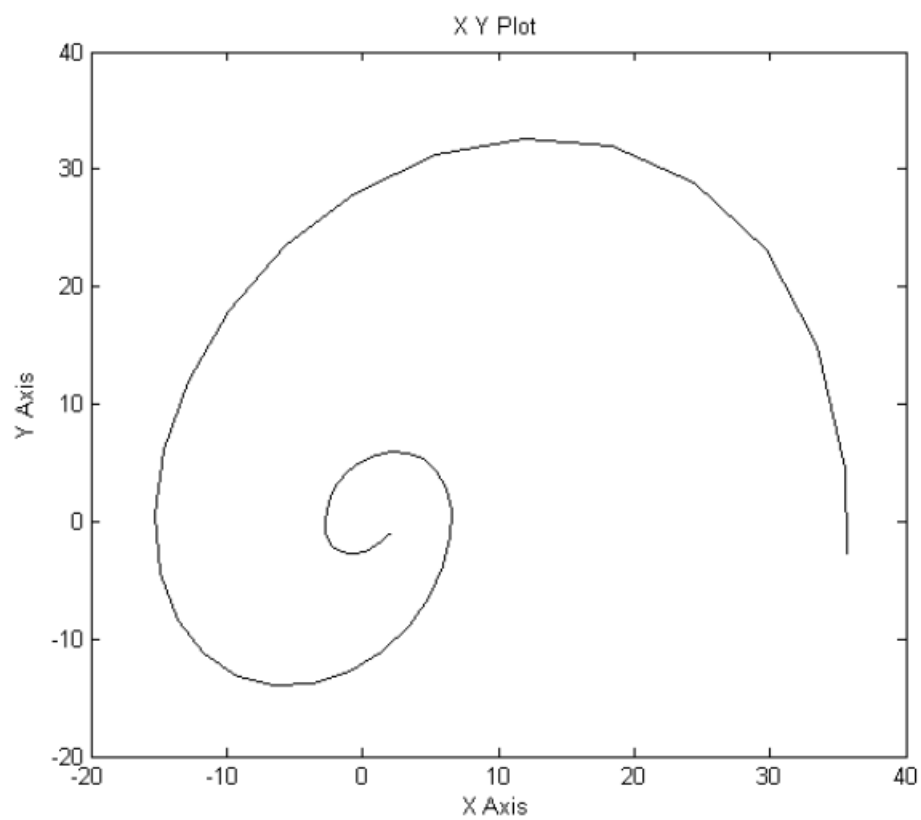


Рис. 4.7. Фазовий портрет системи (приклад 3)

Лекція № 6

Дослідження струмів і напруг послідовного RLC-кола в середовищі Simulink пакетуMatLab

6.1 Общие положения и определения

Электрической цепью называется совокупность устройств и объектов, образующих путь для электрического тока, электромагнитные процессы в которых могут быть описаны с помощью понятии об электродвижущей силе, токе и напряжении.

Цепи делятся на два больших класса:

- линейные, содержащие только элементы с линейными вольтамперными характеристиками
- нелинейные, содержащие элементы с нелинейными характеристиками.

Только линейные элементы подчиняются закону Ома:

$$U = RI; \quad R = \frac{U}{I}; \quad I = \frac{U}{R},$$

где R - сопротивление участка цепи (Ом), U - напряжение на участке (В), I - сила тока (А).

Цепью постоянного тока называется такая, в которой ЭДС, токи и напряжения остаются постоянными по величине и не зависят от времени.

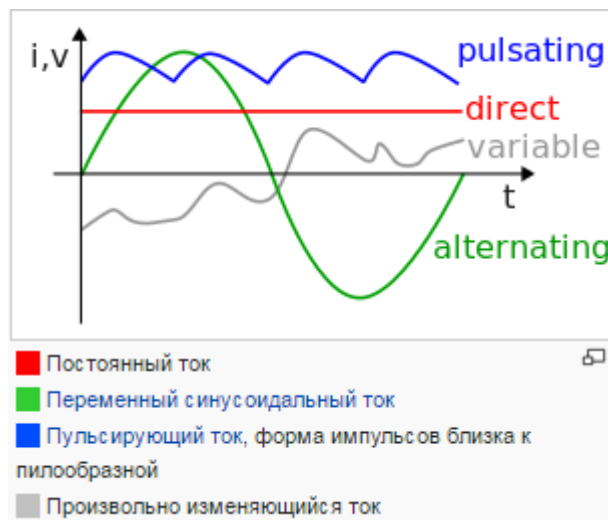


Рис. 6.1.

На рис. 6.1. красным цветом изображен график постоянного тока. По горизонтальной оси отложен масштаб времени t , а по вертикальной – масштаб тока I или электрического напряжения U . Величина постоянного тока I и электрического напряжения U для любого момента времени сохраняется неизменной. Таким образом, постоянный ток – это постоянное направленное движение заряженных частиц.

В каждой точке проводника, по которому протекает постоянный ток, одни элементарные электрические заряды непрерывно сменяются другими, совершенно одинаковыми электрическими зарядами. Несмотря на непрерывное перемещение электрических зарядов вдоль проводника, общее пространственное их расположение внутри проводника как бы остаётся неизменным во времени, или стационарным.

Источником постоянного тока является батарейка, аккумулятор.

Переменный ток — электрический ток, который с течением времени изменяется по величине или направлению или, в частном случае, изменяется по величине, сохраняя своё направление в электрической цепи неизменным (рис. 6.1. зеленый цвет).

Теория переменных токов изучает электрические цепи.

Синусоидальным током называется такой ток, величина и направление которого изменяется в зависимости от времени по закону синуса.

Время, в течении которого синусоид претерпевает полный цикл своего изменения (рис. 6.2) называется периодом (T), размерность периода [$сек$].

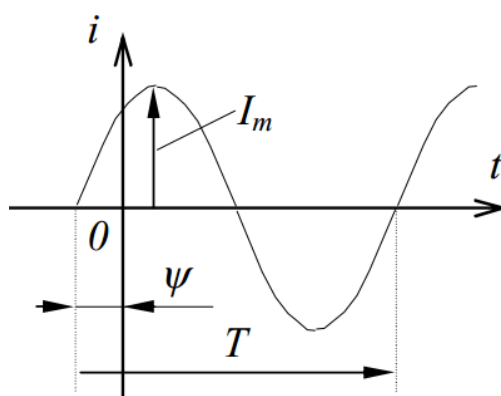


Рис. 6.2. Синусоидальный ток

Параметры переменного тока и напряжения

Величина переменного тока, как и напряжения, постоянно меняется во времени. Количественными показателями для измерений и расчетов применяются их следующие параметры:

Период T - время, в течении которого происходит один полный цикл изменения тока в оба направления относительно нуля или среднего значения.

Частота f - величина, обратная периоду, равная количеству периодов за одну секунду. Один период в секунду это один герц (1 Hz), т.е.

$$[f] = \left[\frac{1}{\text{сек}} \right] = [\text{Герц}] = [\text{Гц}]:$$

$$f = \frac{1}{T}$$

Циклическая частота ω - угловая частота, равная количеству периодов за 2π секунд (рад/сек):

$$\omega = 2\pi f = \frac{2\pi}{T}$$

Обычно используется при расчетах тока и напряжения синусоидальной формы. Тогда в пределах периода можно не рассматривать частоту и время, а исчисления производить в радианах или градусах:

$$T = 2\pi = 360^\circ$$

Начальная фаза ψ - величина угла от нуля ($\omega t = 0$) до начала периода. Измеряется в радианах или градусах. Показана на рис. 6.3. Для синено графика синусоидального тока.

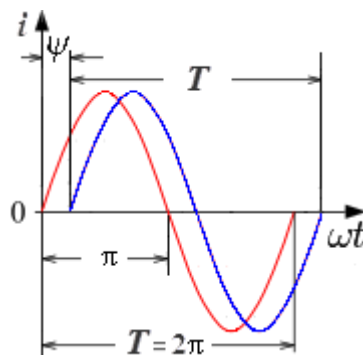


Рис. 6.3.

Начальная фаза может быть положительной или отрицательной величиной, соответственно справа или слева от нуля на графике.

Мгновенное значение – величина напряжения или тока измеренная относительно нуля в любой выбранный момент времени t (сек).

$$i = i(t); \quad u = u(t)$$

Последовательность всех мгновенных значений в любом интервале времени можно рассмотреть как функцию изменения тока или напряжения во времени. Например, синусоидальный ток или напряжение можно выразить функцией:

$$i = I_{amp} \sin(\omega t); \quad u = U_{amp} \sin(\omega t),$$

где I_{amp} и U_{amp} - амплитудные значения тока и напряжения.

Амплитудное значение – максимальное по модулю мгновенное значение за период.

$$I_{amp} = \max|i(t)|; \quad U_{amp} = \max|u(t)|.$$

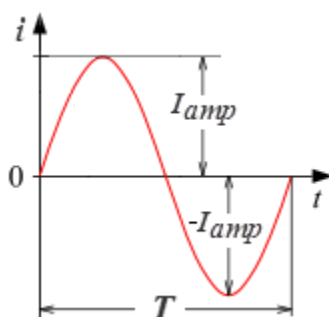


Рис. 6.4.

Может быть положительной и отрицательной в зависимости от положения относительно нуля. Часто вместо амплитудного значения применяется термин амплитуда тока (напряжения) – максимальное отклонение от нулевого значения.

Все существующие электрические цепи, состоят из разных сочетаний четырех основных элементов: активного сопротивления (R), индуктивности или катушка ($L [Гн]$) и выражают в генри (Гн), емкости или конденсатора (C)

и измеряется в фарадах (Ф), усилительно-преобразовательный элемент (диод, транзистор и т.п.).

Первые три элемента в приведенном списке, называются так же пассивными элементами цепи.

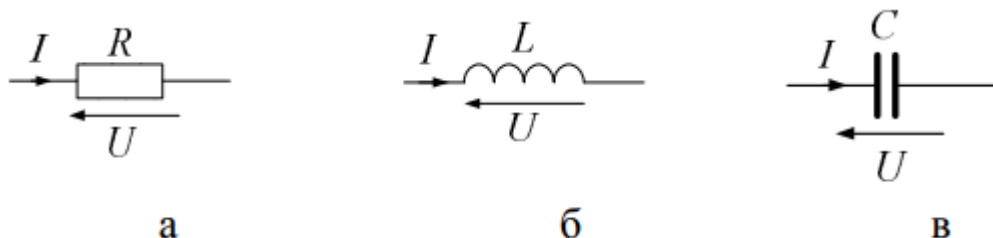


Рис. 6.5. Идеальное активное сопротивление

Полной цепью переменного тока называется такая цепь, в которой активный, индуктивный и емкостной элементы соединены друг с другом последовательно. Часто такие цепи называются цепями *RLC*.

6.2. Основные особенности создания моделей.

Методика создания модели ничем не отличается от методики создания модели на основе базовой библиотеки Simulink. Так же как и для обычной Simulink-модели, необходимо выполнить расстановку блоков на схеме, задать их параметры, соединить блоки и установить параметры расчета модели в целом. Для моделей пакета расширения Sim Power System доступен ускоренный режим расчета и все возможности Simulink, включая набор инструментов Simulink Performance Tools линейный анализ, отладчик и т. д. Однако эти модели имеют и некоторые особенности.

1. Входы и выходы блоков, в отличие от блоков Simulink, не являются направленными, поскольку фактически являются эквивалентами электрических контактов. Таким образом, электрический ток может через вход или выход блока протекать в двух направлениях: как во внутрь блока, так и наружу. Изображение порта на пиктограмме блока представляет собой небольшой квадрат. Эти порты не являются направленными, что

соответствует физической природе сигнала, проходящего через такие порты (электрический ток).

2. Соединительные линии между блоками являются, по сути, моделями электрических проводов, по которым ток может протекать в двух направлениях. В Simulink-моделях информационный сигнал распространяется только в одном направлении — от выхода одного блока к входу другого.

3. Simulink-блоки и Sim Power System-блоки не могут быть непосредственно соединены друг с другом, поскольку имеют сигналы разной природы. Сигнал от S-блока можно передать к SPS-блоку через управляемые источники тока или напряжения, а наоборот — с помощью измерителей тока или напряжения.

4. Несколько линий связи (проводов) могут быть соединены между собой. Для выполнения такого соединения в Sim Power System 3 не требуется использование специальных блоков (соединителей), как это было в предшествующих версиях.

5. При расчете схемы, содержащей нелинейные блоки, следует использовать методы:

- ode15s — многошаговый метод переменного порядка (от 1 до 5), использующий формулы численного дифференцирования;
- ode23tb — неявный метод Рунге-Кутты в начале решения и метод, использующий формулы обратного дифференцирования 2-го порядка, в последующем (они дают наилучшие результаты по быстродействию).

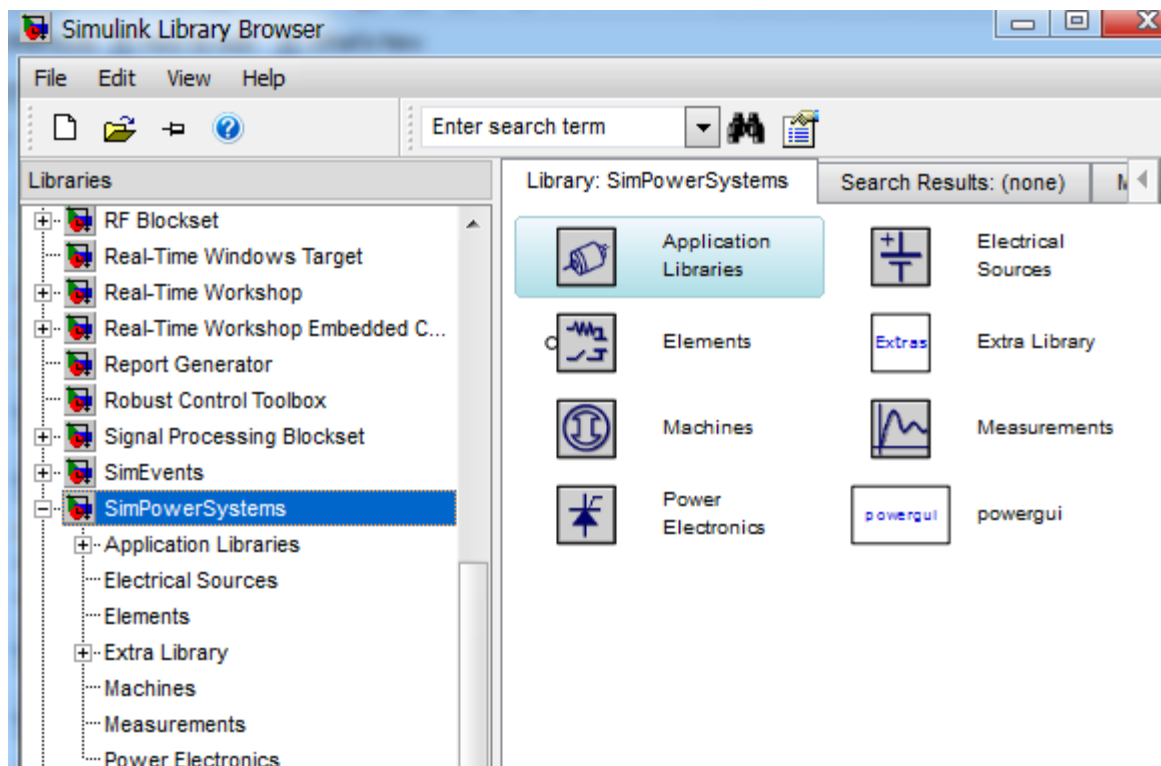


Рис. 6.6. Окно обозревателя библиотеки Simulink с открытым деревом SimPowerSystems

Библиотека SimPowerSystems имеет семь основных разделов (рис.6.6):

- *Electrical Sources* - источники электрической энергии
- *Elements* - электротехнические элементы
- *Power Electronics* - устройства силовой электроники
- *Machines* - электрические машины
- *Measurements* - измерительные и контрольные устройства;
- *Application Library* –библиотека инженерных приложений;
- *Extra Library* - дополнительные библиотеки;

Используя блоки из этих разделов, пользователь в состоянии за короткое время создать полноценную модель достаточно сложной электромеханической системы. В основной библиотеке находится также блок Powergui, используемый для анализа электрических схем.

Данный блок обеспечивает выполнение следующих задач:

- расчет схемы комплексным методом;
- расчет установившегося режима:

- дискретизация модели;
- задание начальных условий;
- инициализация трехфазных схем, содержащих электрические машины, таким образом, чтобы расчет начался с установившегося режима;
- частотный анализ схемы с помощью инструмента Use LTI Viewer.
- определение полного сопротивления (импеданса) цепи;
- выполнение гармонического анализа;
- определение матрицы уравнений модели в пространстве состояний.

Electrical Sources – источники электрической энергии

Эта библиотека содержит неуправляемые и управляемые источники постоянного и переменного напряжения и тока (рис. 6.7). В полях окна настройки параметров каждого блока устанавливаются значения амплитуды напряжения, начальной фазы и частоты.

Поле “Measurements” позволяет подключать блок Multimeter для измерения и наблюдения выходных параметров источника.

Блоки управляемых источников позволяют связать структурные или функциональные схемы основных библиотек “Simulink Blok Library” с элементами библиотеки Sim Power System.

Окна параметров SPS-блоков аналогичны окнам параметров Simulink-блоков.

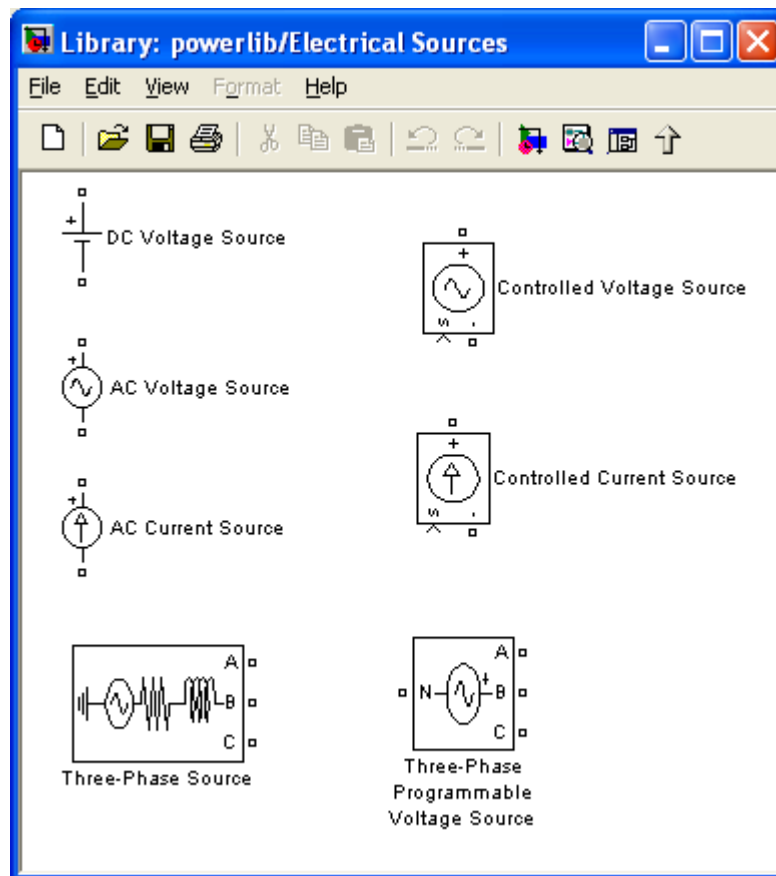


Рис. 6.7. Библиотека Electrical Sources

Elements-электротехнические элементы

На рис. 6.8 представлен состав этой библиотеки, она содержит:

- 1 - Последовательные и параллельные, однофазные и трехфазные пассивные элементы R,L,C, которые могут быть заданы в параметрах этих элементов (Ом, Генри, Фарада “RLC Branch”), а могут быть заданы значениями активной, реактивной индуктивной, либо реактивной емкостной мощностей (“RLC Load”).
- 2 - Однофазные и трехфазные магнитосвязанные цепи (цепи с взаимной индукцией) (Mutual Inductance, Three-phase Mutual Inductance).
- 3 - Линейные однофазные и трехфазные трансформаторы (Linear Transformer, Three-Phase Transformer) и трансформатор с реальным магнитным сердечником, учитывающим его насыщение, (Saturable Transformer).
- 4 - Нелинейный элемент (Surge Arrester), позволяющий сформировать требуемую нелинейную зависимость между входным и выходным сигналами.

5 - Однофазный и трехфазный ключ (Breaker), параметры (сопротивление, индуктивность) которого в открытом состоянии задаются в полях настройки. Там же задается состояние ключа (открыт, закрыт) при нулевом входном сигнале.

6 - Трехфазный двухобмоточный и трехобмоточный трансформаторы (Three-Phase Transformer, Two windings, Three windings).

7 - Блоки, реализующие параметры однофазной и трехфазной линии передачи (PI Section Line, Distributed Parameters Line).

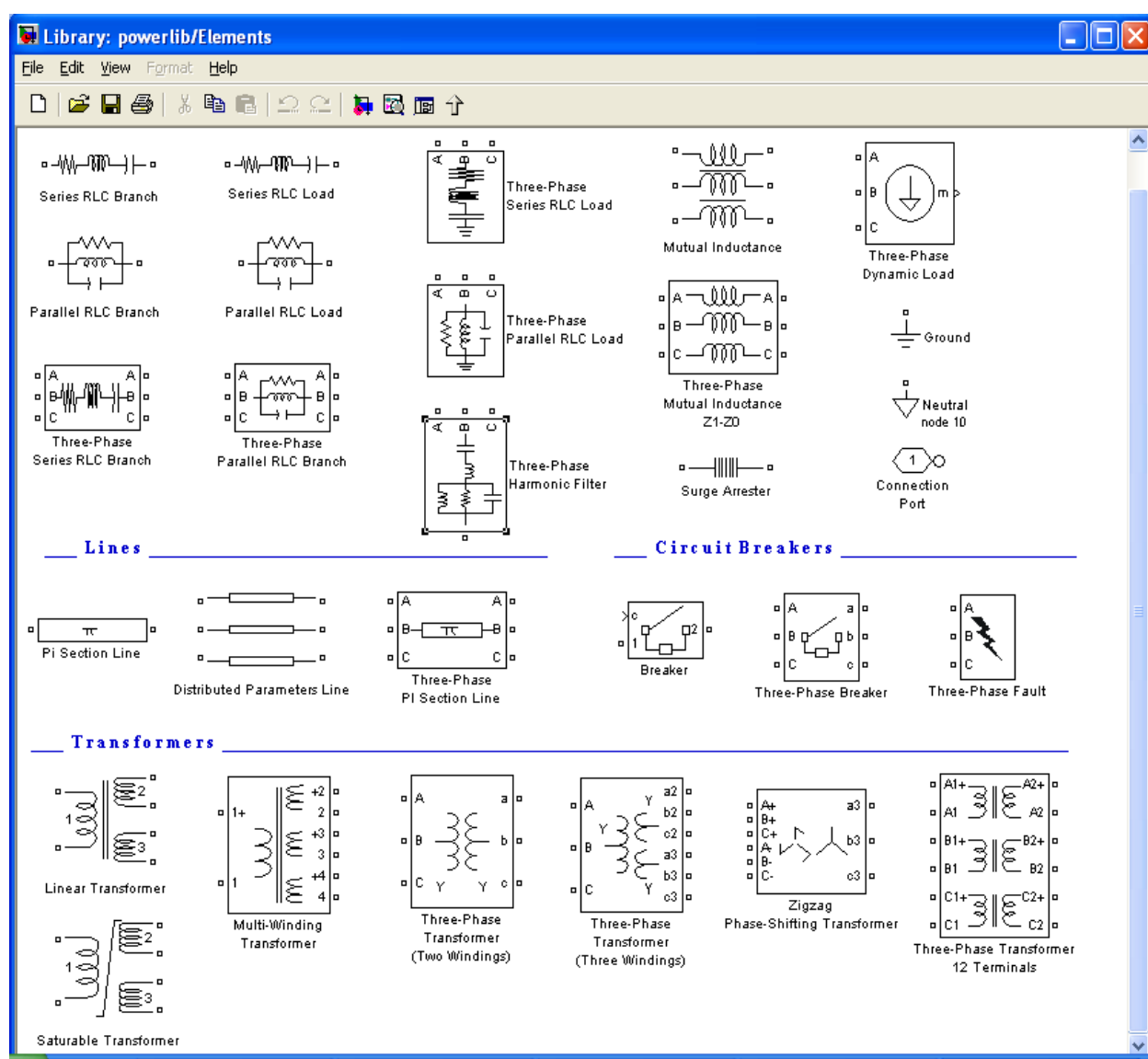


Рис. 6.8. Библиотека Elements

Power Electronics - устройства силовой электроники

В библиотеке устройств силовой электроники (рис. 6.9) находятся модели полупроводниковых приборов (диод, тиристор, запираемый тиристор, IGBT и MOSFET транзисторы), а также универсальный мост, позволяющий моделировать одно-, двух- и трехфазные выпрямительные или инверторные схемы. Кроме того, в библиотеке присутствует модель трехуровневого моста и модель идеального ключа.

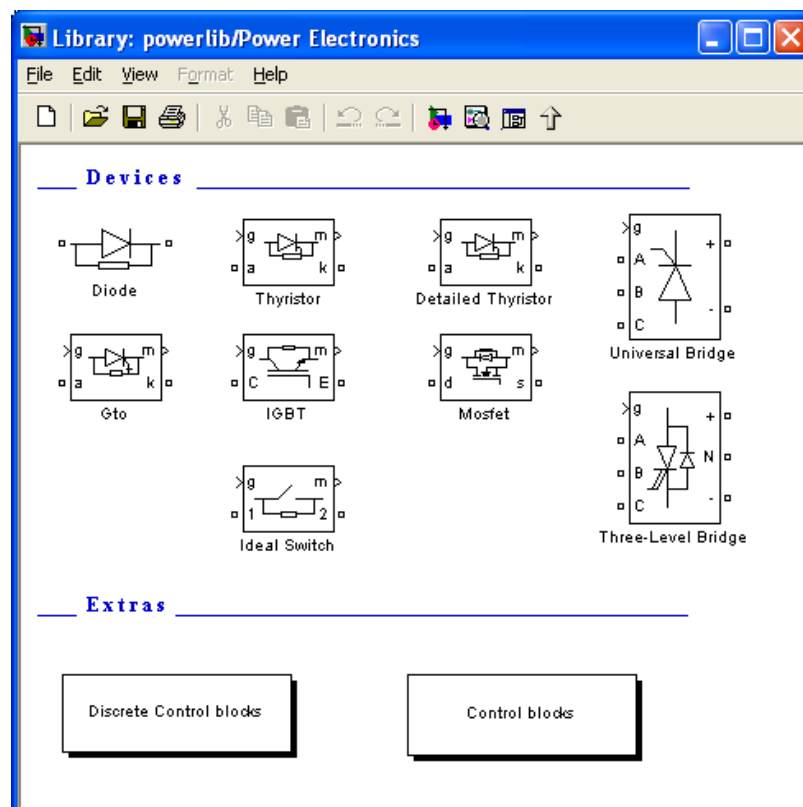


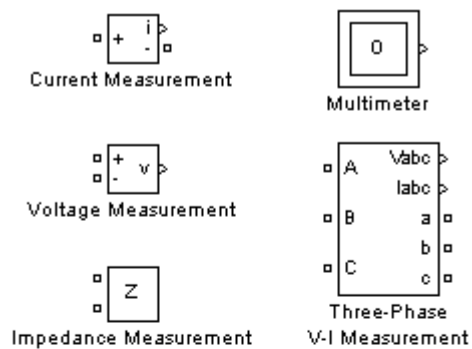
Рис.6.9. Библиотека Power Electronics

Каждый единичный полупроводниковый элемент содержит выход «m.», на котором с помощью измерительных приборов можно посмотреть форму напряжения и тока на элементе и измерить их значения. Все блоки имеют управляющие входы, обозначенные буквой “g”.

В дополнительных библиотеках Extras помещены дополнительные дискретные и непрерывные блоки управления.

Measurements - измерительные и контрольные устройства

Окно библиотеки измерительных устройств показано на рис.6.10.



Extras

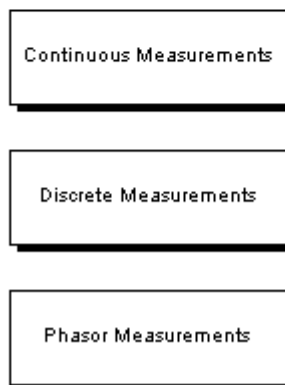


Рис. 6.10. Библиотека измерительных приборов

В библиотеке измерительных устройств находятся блоки измерения мгновенного значения тока и напряжения, блок измерения токов и напряжений в трехфазной системе, измеритель полного сопротивления участка электрической схемы, а также блок Multimeter, позволяющий измерять переменные блоков без непосредственного подключения измерительных устройств. Блок Multimeter аккумулирует переменные тех блоков, для которых в окне их параметров в графе Measurement заданы измеряемые переменные. Выходными сигналами измерительных блоков являются обычные сигналы Simulink, которые можно подать на блоки отображения (Scope, Display) или на вход любого обычного Simulink-блока.

Powerlib Extras – расширенные библиотеки

Среди дополнительных библиотек (рис. 6.11) находится большое количество разнообразных блоков. В частности библиотека трехфазных моделей, библиотека дополнительных измерительных приборов, библиотека устройств управления преобразователями и т.д.

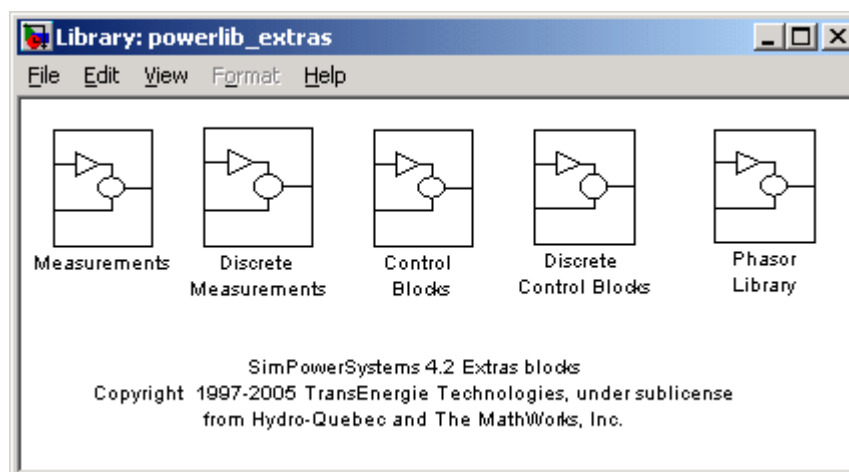


Рис. 6.11. Расширенные библиотеки

Пакет расширения, Sim Power System обеспечивает широкие возможности для расчета электротехнических устройств и систем и в настоящее время может считаться одним из лучших пакетов для моделирования устройств силовой электроники.

Рассмотрим основные элементы силовых полупроводниковых преобразователей и их представление в пакете Sim Power System. В примерах, помещенных ниже, будут одновременно показаны возможности пакетов Simulink, Control System и Sim Power System по визуализации, измерению и исследованию электромагнитных процессов в схемах.

Моделирование электрических цепей в Simulink

Рассмотрим моделирование электрических цепей при этом будем использовать следующие блоки:

DC voltage source – источник постоянного напряжения. В параметрах можно изменять величину генерируемого напряжения (Amplitude), В. Внешний вид блока показан на рис. 6.12, а.

AC voltage source – источник переменного напряжения. Выходное напряжение изменяется по синусоидальному закону. В параметрах изменяется амплитуда напряжения (Amplitude), В; начальная фаза (Phase), эл.град.; частота (Frequency), Гц. Внешний вид блока показан на рис. 6.12, б.

Series RLC Branch – последовательный RLC контур. Представляет собой последовательно соединённые активное, индуктивное и ёмкостное сопротивление. В параметрах можно изменять активное сопротивление (Resistance), Ом; индуктивность (Inductance), Гн; ёмкость (Capacitance), Ф. Внешний вид блока показан на рис. 6.12, в.

Для получения только одного сопротивления необходимо в элементе *Series RLC Branch* исключить остальные два. Исключения сопротивлений осуществляются проставлением для них следующих параметров: для активного сопротивления – «0», для индуктивного сопротивления – «0», для ёмкостного сопротивления – «inf» (англ. *infinity* – бесконечность). На рис. 6.12, г показан элемент, представляющий два последовательно соединённых сопротивления RL. Для получения такого элемента в *Series RLC Branch* заданы параметры $R = 10$; $L = 10$; $C = \text{inf}$.

Ground – заземление. Внешний вид блока показан на рис. 6.12, д, е. Может быть с выходом (рис. 6.12, д) или с входом (рис. 6.12, е).

Voltage measurement – датчик напряжения. На выходе создаёт сигнал, *Current measurement* – датчик тока. На выходе создаёт сигнал, численно равный силе тока, проходящего через него (между входами «+» и «-»). Внешний вид блока показан на рис. 6.12, и.

T-connector – соединитель линий связи. Используется для соединения двух выходов с одним входом. Внешний вид блока показан на рис. 6.12, к.

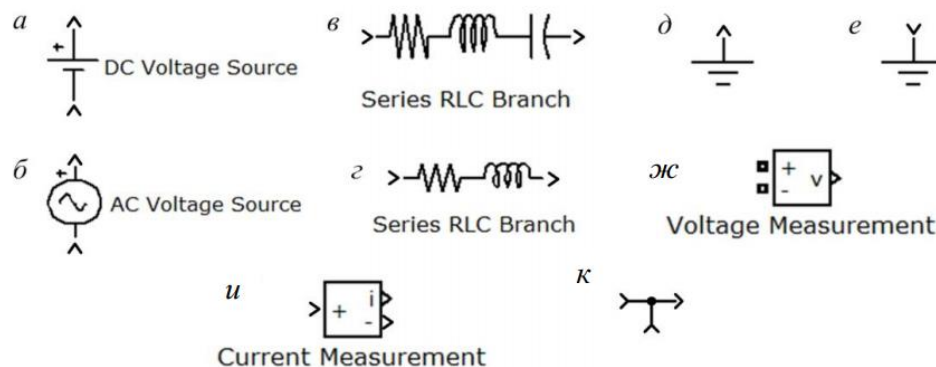
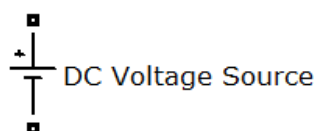


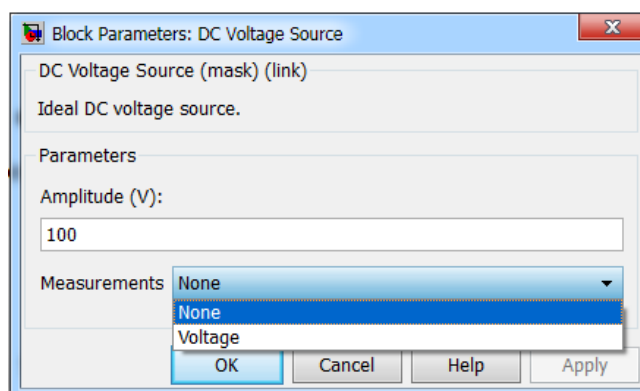
Рис. 6.12. Элементы, необходимые для создания моделей электрических цепей

Идеальный источник постоянного напряжения



Назначение: получение постоянного по уровню напряжения.

Параметр блока:



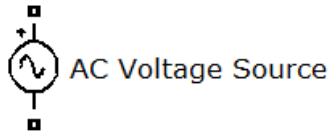
Amplitude (V) – амплитуда (В). Задается уровень выходного напряжения источника. Measurements – измеряемые переменные. Задаются переменные, передаваемые для измерения в блок Multimeter или для наблюдения с помощью блока Scope. Значение параметра выбирается из списка:

None – переменные не отображаются;

Voltage – отображается выходное напряжение источника.

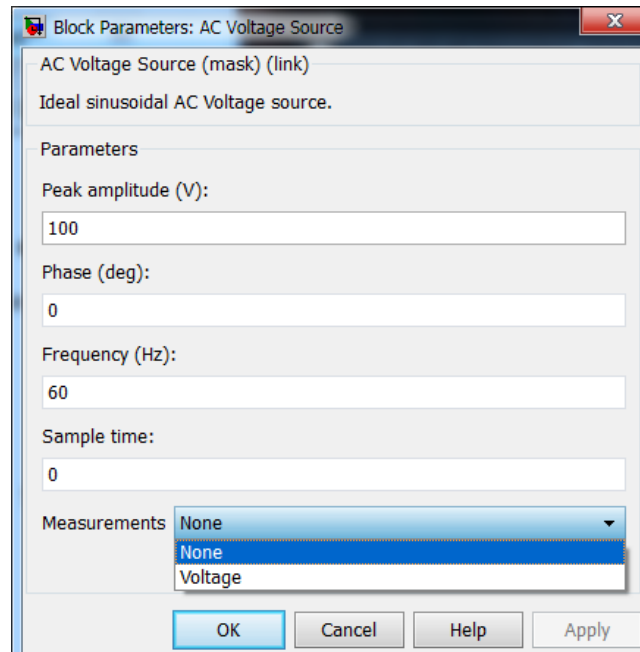
Поскольку блок является идеальным источником напряжения, то его внутреннее сопротивление является нулевым.

Идеальный источник переменного напряжения



Назначение: получение синусоидального напряжения с постоянной амплитудой.

Параметры блока:



Peak Amplitude (V) – амплитуда (В). Задается амплитуда выходного напряжения источника. Phase (deg) – фаза (град). Устанавливается фазовый сдвиг. Frequency (Hz) – частота (Гц). Задается частота переменного напряжения источника. Sample time – шаг дискретизации. С помощью этого параметра задается шаг дискретизации источника при создании дискретных моделей. Measurements – измеряемые переменные. Задаются переменные, передаваемые для измерения в блок Multimeter или для наблюдения с помощью блока Scope.

Значение параметра выбирается из списка:

None – переменные не отображаются;

Voltage – отображается выходное напряжение источника.

Блок является идеальным источником напряжения и имеет нулевое внутреннее сопротивление.

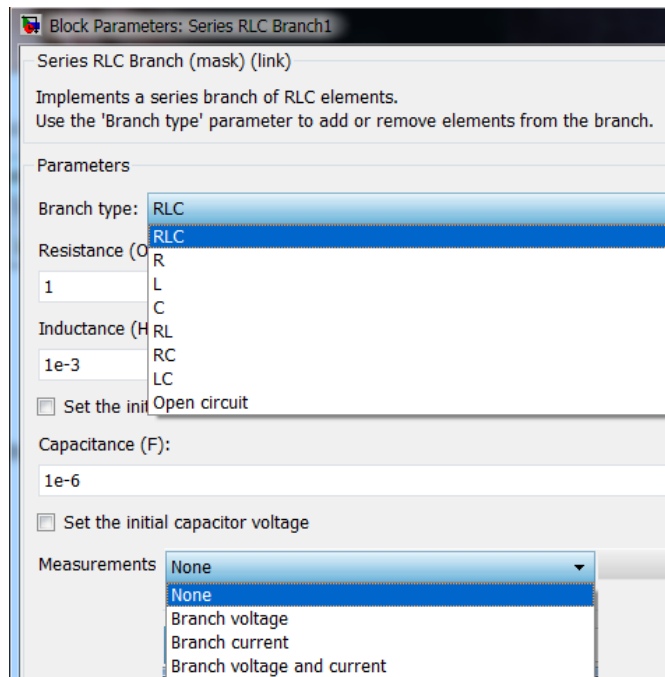
Последовательная RLC - цепь



Series RLC Branch1

Назначение: модель цепи из резистора, индуктивности и конденсатора, соединенных последовательно.

Параметры блока:



Resistance R (Ohms) – сопротивление (Ом). Величина активного сопротивления. Для того чтобы исключить резистор из цепи, значение сопротивления в окне параметров блока нужно задать равным нулю. При этом на пиктограмме блока резистор отображаться не будет. Inductance L (H) – индуктивность (Гн). Величина индуктивности. Для исключения индуктивности из цепи ее значение следует задать в окне параметров равным нулю. При этом на пиктограмме блока индуктивность отображаться не будет. Capacitance C (F) – емкость (Ф). Величина емкости. Для исключения конденсатора из цепи значения емкости следует задать равным ∞ (бесконечность). При этом конденсатор на пиктограмме блока отображаться не будет. Measurements измеряемые переменные. Задаются переменные, представляемые для измерения в блок Multimeter или для отображения с помощью блока Scope. Значение параметра выбирается из списка:

None – нет переменных для отображения;

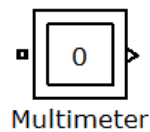
Branch voltage – напряжение на зажимах цепи;

Branch current – ток цепи;

Branch voltage and current – напряжение и ток цепи.

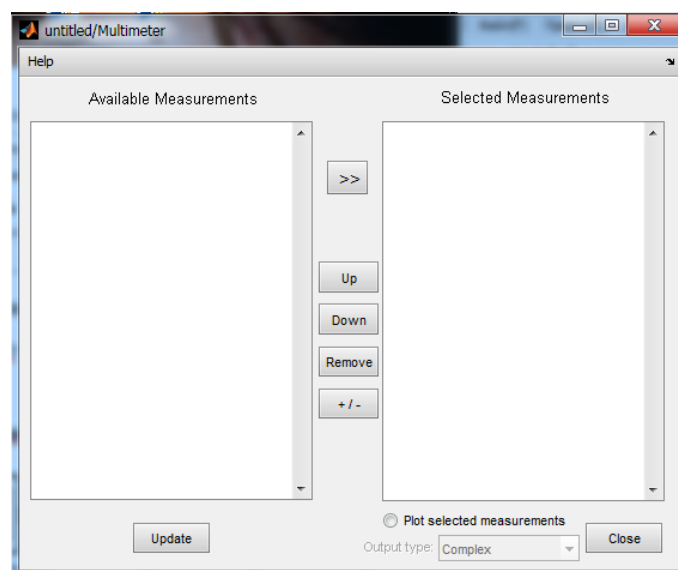
Отображаемым сигналам в блоке Multimeter присваиваются обозначения: I_b – ток цепи; U_b – напряжение цепи.

Мультиметр



Назначение: измеряет токи и напряжения блоков библиотеки SimPowerSystem, для которых в их окне настройки параметров устанавливается параметр Measurements – «измеряемые переменные».

Параметры блока:



Available Measurements – переменные (в левом окне), доступны для измерения. Это токи и напряжения блоков схемы, для которых в окне настройки параметров блока установлен параметр Measurements (измеряемые переменные). Обновления списка переменных можно выполнить с помощью клавиши Update, но предварительно надо внести изменения в окна настройки параметров блока. Selected Measurements – измеряемые переменные (в правом

окне). Указываются переменные, которые будут передаваться на выход блока Multimeter. Для управления списком измеряемых переменных можно использовать следующие клавиши (между окнами):

>> - добавить выделенную переменную в правый список;

Up – передвинуть вверх выделенную переменную в правом списке;

Down – передвинуть вниз выделенную переменную в правом списке;

Remove – удалить выделенную переменную из правого списка;

+/- - изменить знак выделенной переменной.

Output signal – выходной сигнал. Вид выходного сигнала блока. Выбор значения параметра возможен при условии, что с помощью блока Powergui установлен режим расчета на переменной токе (Phasor simulation). В этом случае значение параметра выбирается из списка:

Magnitude – амплитуда (скалярный сигнал);

Complex – комплексный сигнал;

Real-Imag – вектор, состоящий из двух элементов – действительной и мнимой составляющих сигнала;

Magnitude-Angle – вектор, состоящий из двух элементов – амплитуды и аргумента сигнала.

Блок может использоваться для измерения напряжения и токов вместо обычных измерителей Current Measurements и Voltage Measurements. Выходным сигналом блока является вектор всех сигналов, соответствующих измеряемым переменным.

Пример 6.1. Представить модель простой электрической цепи постоянного тока, содержащей источник постоянного напряжения и активное сопротивление. Источник постоянного напряжения генерирует напряжение величиной $U = 10$ В, активное сопротивление составляет $R = 3$ Ом. Время моделирования 0-2 с, метод расчета *ode23s*.

Расчет модели:
$$I = \frac{U}{R} = \frac{10}{3} \approx 3,333 \text{ А.}$$

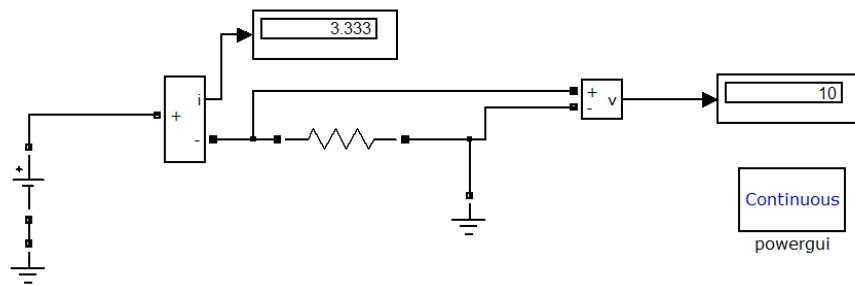


Рис. 6.13. Модель простой электрической цепи постоянного тока

Пример 6.2. Смоделировать электрическую цепь переменного тока, содержащую источник переменного напряжения и последовательно соединенные активное, индуктивное и емкостное сопротивления. Параметры источника напряжения: амплитуда – 10 В, частота 50 Гц, фаза – 0. Параметры RLC: $R=0,1$ Ом, $L=0,001$ Гн, $C=0,15$ Ф. Время моделирования 0,9-1 с; метод расчета ode23s; максимальный шаг интегрирования 0,0001.

Результаты расчета электрической цепи:

$$\dot{I}_m = \frac{\dot{U}_m}{Z} = \frac{10}{0,1 + \frac{1}{j314,16 \cdot 0,15} + j314,16 \cdot 0,001} = 10,437 - j30,574 = 32,3e^{-71,2} \text{ А.}$$

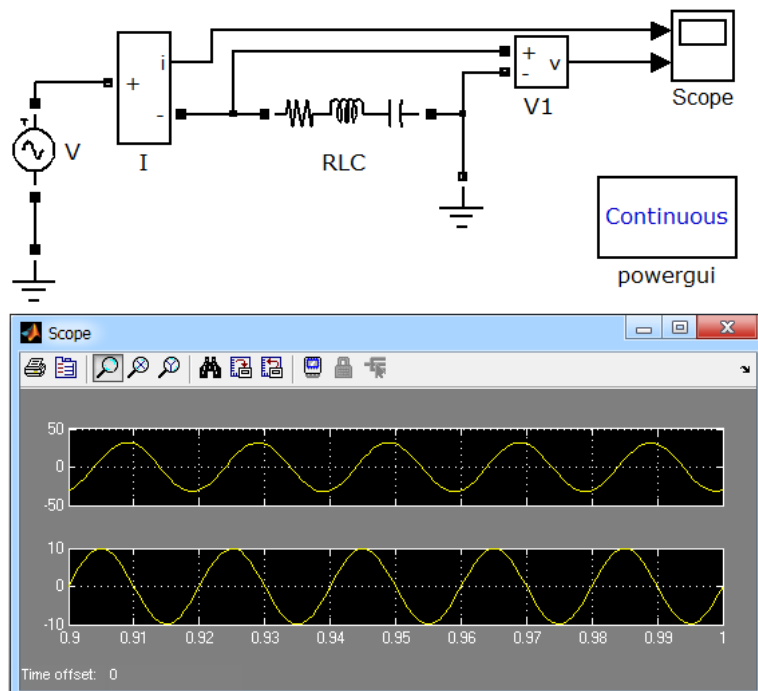


Рис. 6.14. Модель простой электрической цепи переменного тока и результаты ее расчета на осциллографе

Пример 6.3. Смоделировать схему включения двух источников постоянного напряжения $V_1=200\text{ В}$, $V_2=300\text{ В}$. На общий импеданс ($R=50\text{ Ом}$, $L=1\text{ мкГ}$).

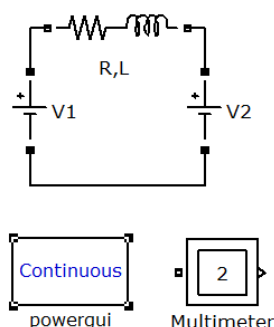
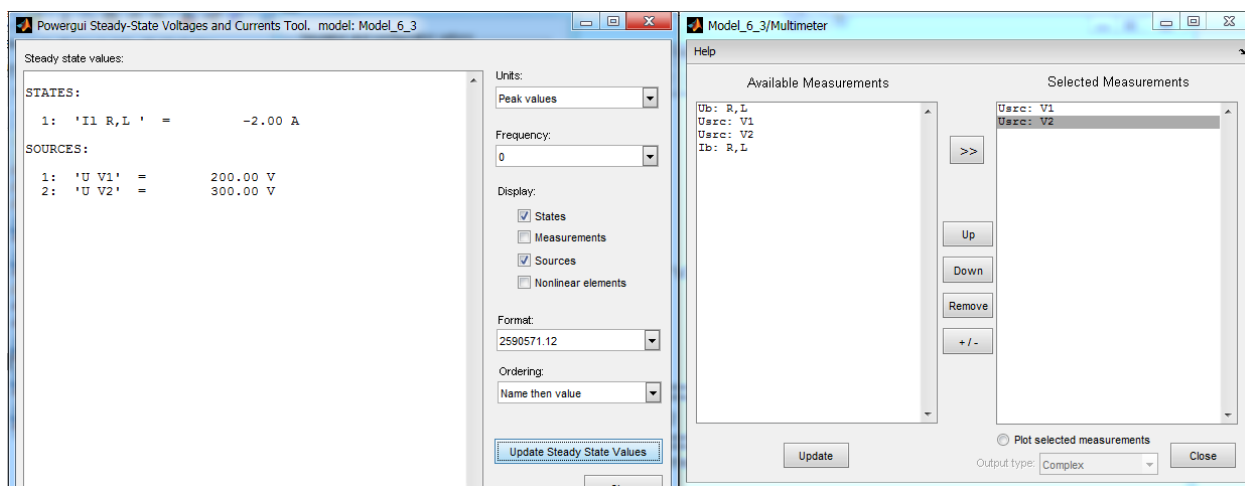


Рис. 6.15. Модель электрической схемы



6.16. Окна блоков Powergui и Multimeter с результатами измерений

Пример 6.4. Смоделировать электрическую схему последовательного включения двух источников синусоидального напряжения с параметрами $u_1 = 100\sin 2\pi \cdot 50t$ и $u_2 = 100\sin\left(2\pi \cdot 50t + \frac{\pi}{3}\right)$ на катушку индуктивности с параметрами $R=3\text{ Ом}$, $L=0,01\text{ Гн}$. Графическое изображение напряжение и ток в катушке показать на осциллографе. Для наблюдения величин, измеренных блоком Multimeter, необходимо их перенести из поля Available Measurements в поле Selected Measurements.

Примечание. Блок Scope из основной библиотеки имеет черное поле экрана. Для того, чтобы иметь возможность редактировать цвет экрана, цвет и толщину линий осциллограммы необходимо в рабочем пространстве MatLab набрать и выполнить команды:

```
set(0,'ShowHiddenHandles','On')
set(get,'menubar','figure')
```

В этом случае в окне блока появится панель инструментов, позволяющая редактировать экран и осциллограммы.

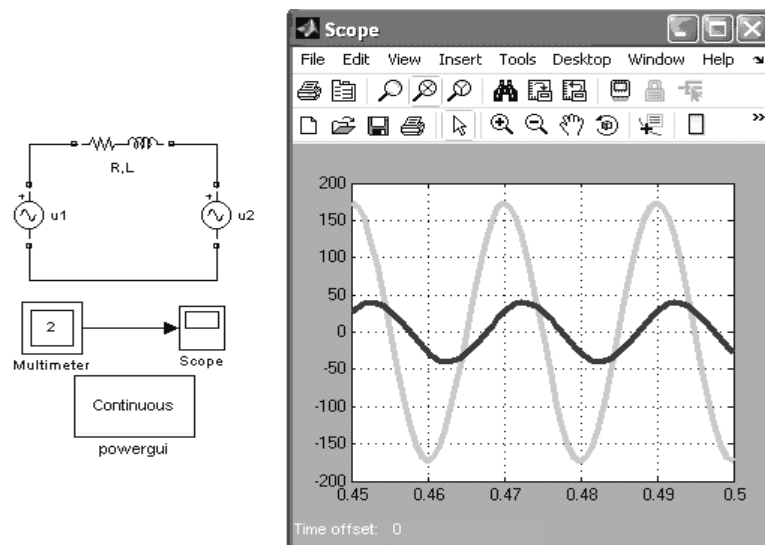


Рис. 6.17. Модель последовательного включения двух источников синусоидального напряжения на общую нагрузку

Измерение блоком Powergui амплитуды и начальные фазы на всех элементах модели определяются в окне этого блока:

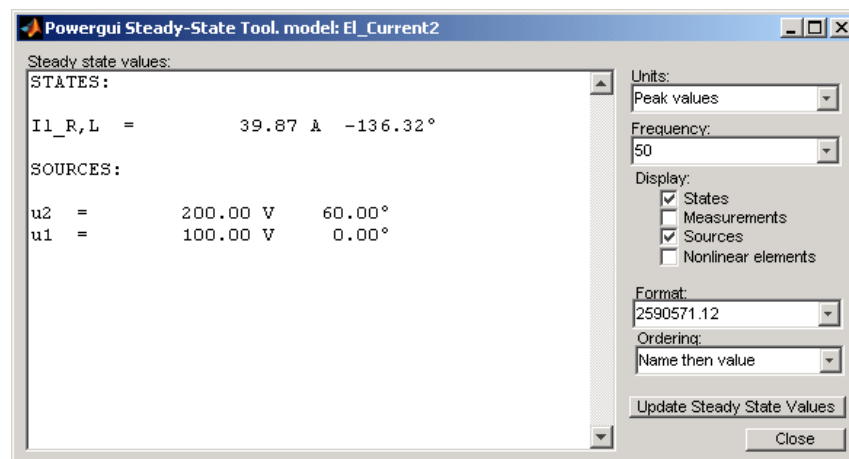


Рис. 6.18. Окно блока Powergui

Controlled Voltage Source. Управляемый источник напряжения.

Controlled Current Source. Управляемый источник тока

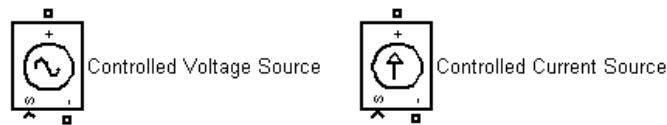


Рис. 6.19. Пиктограммы управляемых источников напряжения и тока

Пиктограммы блоков показаны на рис. 6.19. Блоки вырабатывают напряжение (ток) в соответствии с сигналом управления.

Окна задания параметров блоков показаны на рис. 6.20.

В полях окна задаются: Initialize. При установке флажка выполняется инициализация источника с заданными начальными параметрами - амплитудой, фазой и частотой.

Source type - тип источника указывается, если требуется инициализация источника. Если инициализация источника не задана, то параметр недоступен. Значение параметра выбирается из списка:

- AC - источник переменного напряжения (тока),
- DC - источник постоянного напряжения (тока).

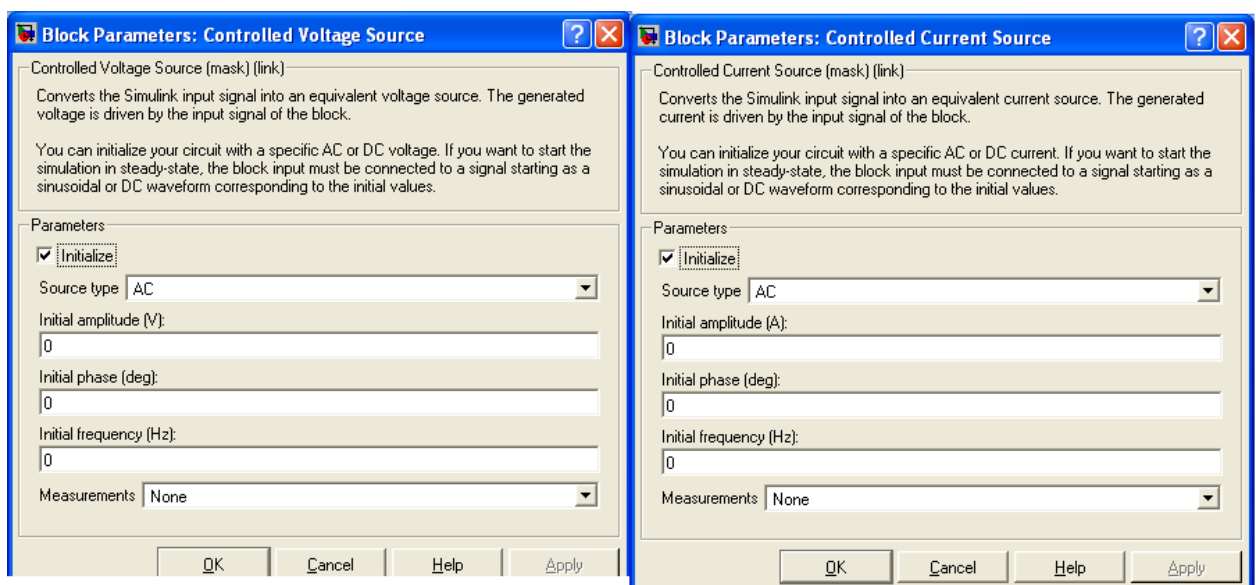


Рис. 6.20. Окна настройки параметров управляемых источников

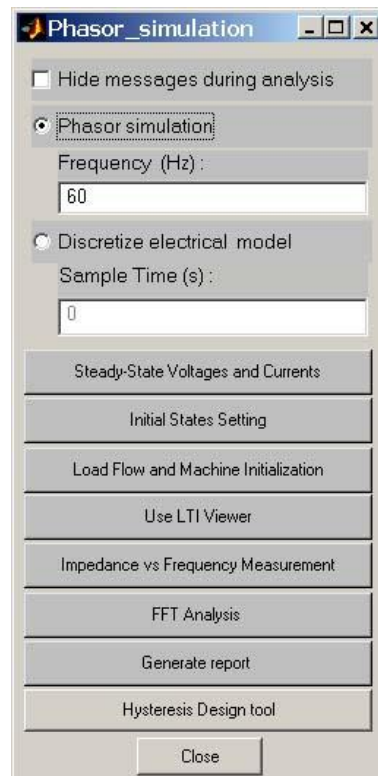
Initial amplitude - начальное значение выходного напряжения (тока) источника. Параметр доступен, если задана инициализация источника. Phase (deg) - начальная фаза (град). Параметр доступен, когда источник инициализируется как источник переменного напряжения. Initial frequency (Hz)- начальная частота источника (Гц). Параметр доступен, когда источник инициализируется как источник переменного напряжения. Measurements- измеряемые переменные. Параметр позволяет выбрать, передаваемые в блок Multimeter, переменные.

Powergui - графический интерфейс пользователя пакета моделирования энергетических систем

Назначение: Блок является инструментом графического интерфейса пользователя и обеспечивает решение следующих задач:

- расчет схемы комплексным методом,
- расчет установившегося режима,
- дискретизация модели,
- задание начальных условий,
- инициализация трехфазных схем содержащих электрические машины, таким образом, чтобы расчет начался с установившегося режима,
- анализ схемы с помощью инструмента Simulink LTI-Viewer,
- определение полного сопротивления (импеданса) цепи,
- выполнение гармонического анализа,
- создание отчета,
- создание файла характеристик намагничивания для модели нелинейного трансформатора.

Окно задания параметров:



Параметры блока:

Hide messages during analysis [Скрывать сообщения при проведении анализа]. Если флажок установлен, то подавляется вывод сообщений в командном окне MATLAB при выполнении расчетов.

Phasor simulation [Расчет схемы комплексным методом]. При установленном флажке выполняется расчет схемы комплексным методом. При этом необходимо задать частоту источников в графе Frequency.

Frequency (Hz) [Частота (Гц)]. Частота источников при расчете схемы комплексным методом. При других видах анализа параметр является недоступным.

Discretize electrical model [Выполнить дискретизацию модели]. При установленном флажке выполняется дискретизация модели. При этом необходимо задать шаг дискретизации в графе Sample time.

Sample time (s) [Шаг дискретизации]. Параметр является доступным, если задан режим дискретизации модели. При этом на пиктограмме блока будет показана величина этого параметра.

Steady State Voltages and Currents [Установившиеся значения напряжений и токов]. Расчет установившихся значений переменных. При нажатии на кнопку открывается окно в котором будут показаны соответствующие значения.

Initial states Setting [Установка начальных значений]. При нажатии на кнопку открывается окно, в котором отображаются начальные значения переменных. Эти значения можно изменять. Новые значения используются при расчете переходных процессов.

Load Flow and Machine Initializations [Инициализация схем содержащих электрические машины].

Use LTI Viewer [Использование LTI Viewer]. Применение инструмента Simulink LTI Viewer для анализа схемы.

Impedance vs Frequency Measurements [Определение импеданса цепи]

FFT Analysis [Гармонический анализ].

Generate Report [Создание отчета].

Hysteresis Design Tool [Инструмент расчета характеристики намагничивания].

Лекція № 8

Моделювання механічних систем у середовищі Simulink

8.1 Общие положения и определения

Для создания модели механической системы необходимо вывести уравнения движения и составить Simulink-модель из стандартных блоков Simulink.

Рассмотрим пример составления модели маятника. Составим расчетную схему и уравнения движения маятника.

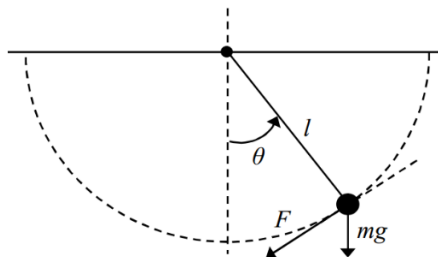


Рис. 8.1. Расчетная схема маятника

Груз некоторой массы прикреплен к абсолютно упругому и навесному стержню длины l , стержень крепится к шарнирной опоре, θ - угол отклонения маятника от вертикального положения, g - ускорение свободного падения, t - время, μ - коэффициент трения при движении маятника в некоторой среде.

Математически модель маятника описывается:

$$l \frac{d^2 \theta}{dt^2} = \mu \frac{d\theta}{dt} + g \sin(\theta) = 0$$

Пусть $y = \theta$; $y' = \theta'$, тогда перепишем наше уравнение $y'' = -\frac{\mu}{l} y' - \frac{g}{l} \sin y$.

Исходя из вышеизложенного уравнения построим схему модели маятника в Simulink и графически покажем затухающий колебательный момент.

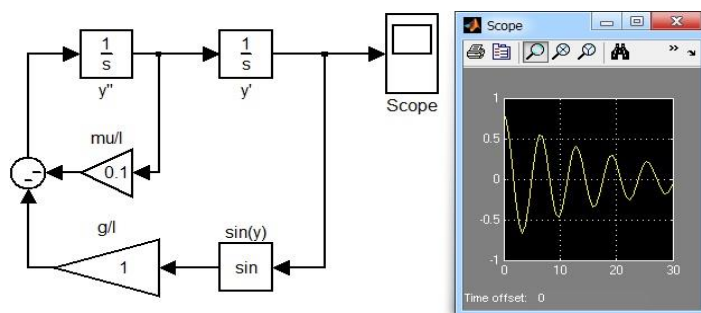


Рис. 8.2. Simulink модель работы маятника

SimMechanics - пакет расширения системы Simulink для физического моделирования. Его цель - техническое проектирование и моделирование механических систем (в рамках законов теоретической механики). SimMechanics позволяет моделировать поступательное и вращательное движения в трех плоскостях. SimMechanics содержит набор инструментов для задания параметров звеньев (масса, моменты инерции, геометрические параметры), кинематических ограничений, локальных систем координат, способов задания и измерения движений. SimMechanics позволяет создавать модели механических систем подобно другим Simulink-моделям в виде блок-схем. Встроенные дополнительные инструменты визуализации Simulink позволяют получить упрощенные изображения трехмерных механизмов как в статике, так и в динамике. При использовании SimMechanics, вместо того, чтобы выводить математическую модель системы, составляется представление, описывающее компоненты механической системы (тела, шарниры и т.д.). В нашем случае

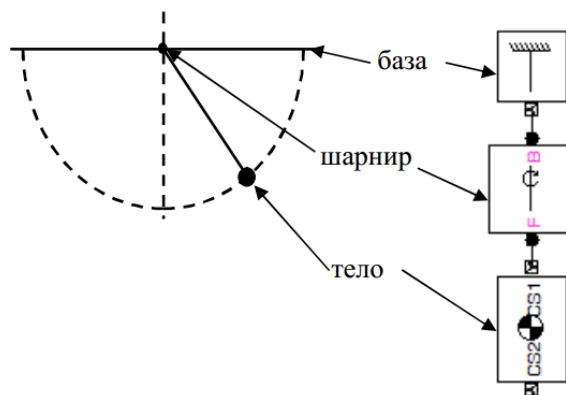


Рис. 8.3. Представление механической системы в SimMechanics на примере маятника

Блоки в библиотеке SimMechanics имеют специальные имена, которые точно описывают их назначение. Например, вращательный шарнир в маятнике называется Revolute, подвижное звено (тело) – Body отображают составные части механизма и неподвижное окружение механизма (основание). неподвижное звено (земля) – Ground. Узловые блоки шарниров Joints изображают степени свободы одной части механизма относительно другой или основания. Блоки стационарных и нестационарных связей Constraints & Drivers ограничивают или запускают движения частей механизма

относительно друг друга. Блоки возбудителей механизма Actuators задают приложенные усилия, движения, меняющиеся массу и инерцию или первоначальные условия для частей механизма, узлов шарниров и связей. Блоки силовых упруговязких элементов Force Elements моделируют усилия между частями механизма. Сенсорные блоки Sensors измеряют усилия или движения частей механизма, узлов шарниров и связей. Встроенные инструментальные средства визуального наблюдения SimMechanics позволяют запустить 3-D анимацию объекта в процессе моделирования, используя графические возможности системы MATLAB.

Чтобы построить SimMechanics-модель маятника нужно перенести соответствующие блоки в окно модели и соединить их вместе. Блоки SimMechanics имеют специальные порты, которые соединяются линиями. Эти соединительные линии не передают данные, а представляют собой механические связи между элементами системы. Соединительные линии и специальные порты не могут напрямую подключаться к стандартным блокам Simulink и линиям. Датчики позволяют измерять положение скорость и ускорение элементов механической системы. В модели маятника, можно измерить угол (AP) и угловую скорость (AV) на вращательном шарнире (рис. 8.4).

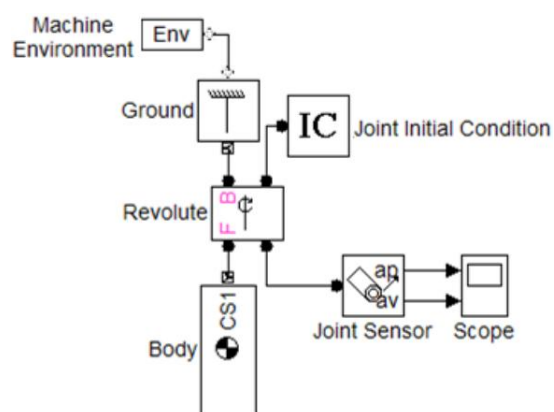


Рис. 8.4. SimMechanics – модель маятника

Моделирование SimMechanics-моделей отличается от моделирования обычных Simulink-моделей. При запуске SimMechanics анализирует механическую систему для определения топологии и геометрии механизма.

Во время моделирования внешние силы и моменты, прикладываемые к механизму, интегрируются, и состояние механической системы обновляется. Так как модель может содержать ограничения, решатель проверяет все элементы механизма в пределах допустимых отклонений.

8.2. Библиотека блоков пакета SimMechanics

Также, как все остальные библиотеки Simulink, библиотека пакета SimMechanics представляет собой набор блоков в виде графических пиктограмм с оригинальными названиями на английском языке. Для их просмотра, выбора и перетаскивания мышью в окно создаваемой Simulink-модели (в дальнейшем S-модели) служит окно браузера библиотек Simulink (рис. 8.5).

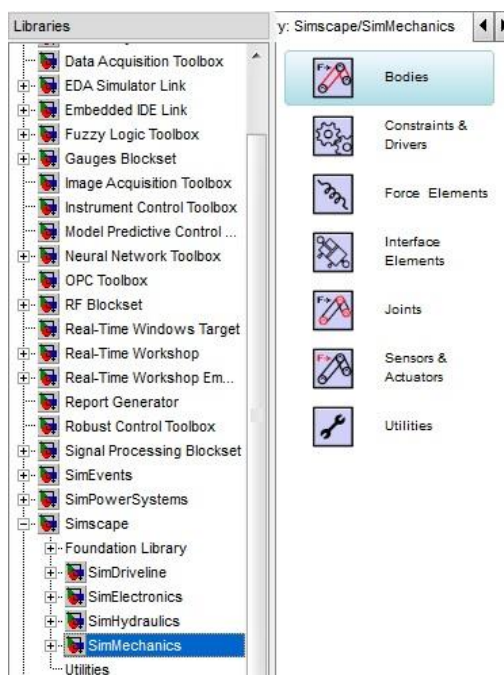


Рис. 8.5. Окно браузера библиотек Simulink, разделы пакета SimMechanics

Как видно на рис. 8.5, в библиотеке SimMechanics всего семь разделов: Bodies; Constraints & Drivers; Force Elements; Interface Element; Joints; Sensors & Actuators; Utilities.

Каждый раздел содержит блоки определенной группы. Рассмотрим их более подробно

1. Блоки твердых тел (Bodies)

Данный раздел библиотеки SimMechanics можно считать основополагающим. С него необходимо начинать при создании любой механической модели, хотя он содержит всего три блока (рис. 3):

- Body;
- Ground;
- Machine Environment.

1. Блок **Body** представляет собой твердое жесткое тело (отдельное звено механизма, движения которого моделируются) с определенными пользователем параметрами.

В качестве задаваемых параметров выступают:

- масса тела (Mass) (рис. 8.6), которая может быть выражена в различных единицах (имеется выпадающий список, по умолчанию стоят килограммы);

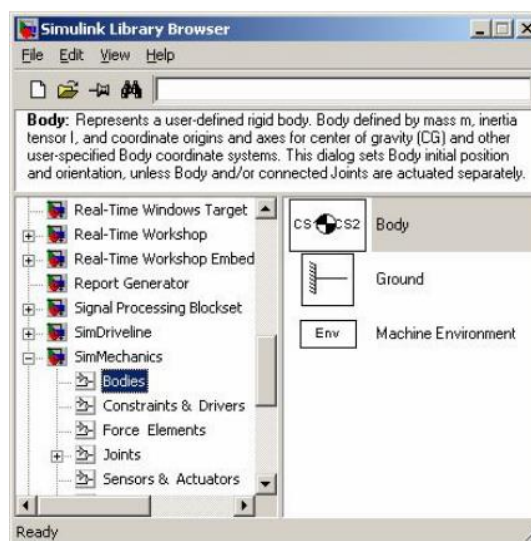


Рис. 8.6. Блоки твердых тел Bodies

- тензор инерции тела относительно его центра масс (Inertia), представляющий собой матрицу размером 3×3:

$$H = \begin{bmatrix} J_{ix} & 0 & 0 \\ 0 & J_{iy} & 0 \\ 0 & 0 & J_{iz} \end{bmatrix},$$


где J_{ix} , J_{iy} , J_{iz} -осевые моменты инерции тела относительно осей собственной локальной системы координат, связанной с его центром масс. Для ряда

наиболее распространенных симметричных тел простой геометрической формы формулы для вычисления элементов тензора инерции приведены в табл. 8.1;

Элементы тензора инерции ряда тел простой формы

Форма тела	Формулы для вычисления		
	J_{ix}	J_{iy}	J_{iz}
Тонкий стержень длиной L вдоль оси Z	$1/12(m \cdot L^2)$	$1/12(m \cdot L^2)$	0
Сфера радиусом R	$2/5(m \cdot R^2)$	$2/5(m \cdot R^2)$	$2/5(m \cdot R^2)$
Цилиндр радиусом R и высотой h с осью вращения Z	$\frac{1}{4}m \left(R^2 + \frac{1}{3}h^2 \right)$	$\frac{1}{4}m \left(R^2 + \frac{1}{3}h^2 \right)$	$1/2(m \cdot R^2)$
Прямоугольный параллелепипед со сторонами a , b и c вдоль осей X , Y и Z соответственно	$\frac{1}{12}m(b^2 + c^2)$	$\frac{1}{12}m(a^2 + c^2)$	$\frac{1}{12}m(a^2 + b^2)$
Конус базового радиуса R и высоты h с осью вращения Z	$\frac{1}{4}m \left(\frac{3}{5}R^2 + h^2 \right)$	$\frac{1}{4}m \left(\frac{3}{5}R^2 + h^2 \right)$	$3/10(m \cdot R^2)$
Эллипсоид с размерами a , b и c вдоль осей X , Y и Z соответственно	$\frac{1}{5}m(b^2 + c^2)$	$\frac{1}{5}m(a^2 + c^2)$	$\frac{1}{5}m(a^2 + b^2)$

- декартовы координаты характерных точек, связанных с телом (Origin position vector [x y z]). В частности, обязательно должны быть указаны координаты центра тяжести тела CG и, при необходимости, координаты произвольного числа других характерных точек CS1, CS2, CS3, ... CSN, таких как центры шарниров, связанных с данным телом, или точки приложения внешних сил и моментов.

Для добавления, удаления и перемещения строк, каждая из которых связана с определенной точкой, служит группа кнопок .

Для того, чтобы добавить или убрать изображение входа/выхода определенной точки на пиктограмме блока Body в окне модели, используется установка или сброс флажка в колонке Show port.

В колонке Translated from origin of указывается имя системы координат (характерной точки), от начала которой отсчитываются координаты текущей характерной точки. Как минимум одна характерная точка (любая, в том числе это может быть и центр тяжести тела CG) должна быть задана в системе координат WORLD либо в системе координат ADJOINING. Это необходимо, чтобы связать данное тело с соседними неподвижными или подвижными телами.

Система координат WORLD – это инерциальная неподвижная система, связанная с Землей, а система координат ADJOINING связана с шарниром, присоединенным к телу (с той частью шарнирного сочленения, которая жестко связана с рассматриваемым телом).

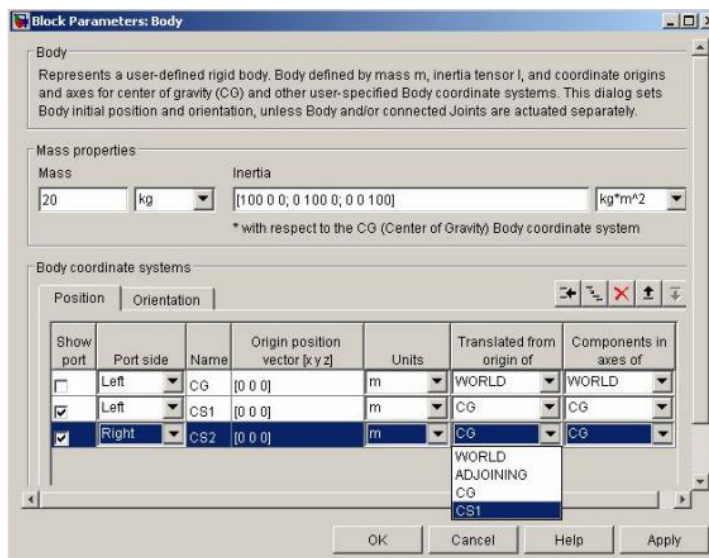


Рис. 8.7. Настройка параметров блока Body

Каждой точке, где тело через шарнир соединяется с другим телом, должна быть поставлена в соответствие своя отдельная система координат CS. Это же касается и точек приложения к телу внешних сил и моментов (при их наличии).

- векторы углов поворота систем координат (на вкладке Orientation). По умолчанию все углы поворота всех систем приняты нулевыми. Изменение их в большинстве случаев не требуется. Координатные оси любой системы координат в SimMechanics по умолчанию расположены так, как показано на рис. 8.8, что соответствует правой Эйлеровой системе координат (Euler X-Y-Z).

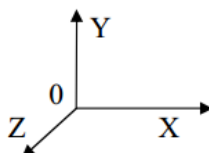


Рис. 8.8. Расположение систем координат

2. Блок **Ground** представляет собой неподвижную стойку (основание), жестко связанную с абсолютной инерциальной системой координат Земли. Наличие хотя бы одного этого блока обязательно в любой механической модели отдельного механизма (иначе при запуске моделирования будет выдана ошибка). В окне настройки этого блока задается вектор координат $[X \ Y \ Z]$ одной неподвижной точки механизма относительно глобальной инерциальной системы координат WORLD. К механическому входу/выходу блока Ground, соответствующему заданной неподвижной точке, должен быть присоединен в окне модели блок шарнирного сочленения (из раздела Joints), а затем блок Body, опять блок из раздела Joint, блок Body и т.д., для того, чтобы стало возможным движение последующих тел Body.

3. Блок **Machine Environment** представляет собой настроечный блок параметров механической среды моделирования для машины (механизма). Блок Machine Environment связан с механическим блоком Ground (рис. 8.9).

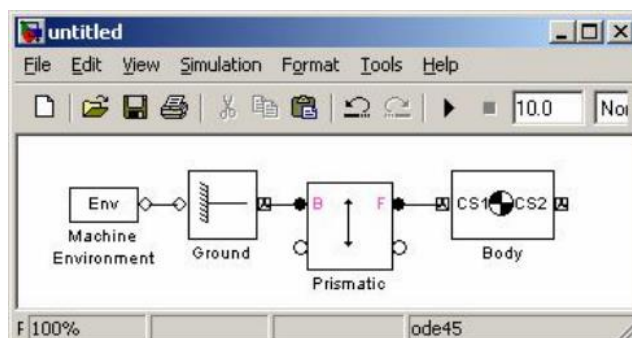


Рис. 8.9. Модель механической системы с блоком Machine Environment

При двойном щелчке мышью по данному блоку открывается окно установки параметров механического моделирования (рис. 8.10). Среди задаваемых параметров на четырех вкладках данного блока имеются: вектор сил тяжести вида $[X \ Y \ Z]$, размерность механизма (2-мерная либо 3-мерная схема), способ анализа, тип решающего устройства ограничения, допуски линеаризации, установка/снятие визуального наблюдения.

По умолчанию вектор сил тяжести установлен как для механизма, расположенного на горизонтальной опорной поверхности с нулевыми углами наклона относительно гравитационной вертикали, и имеет вид $[0 \ -9.81 \ 0]$ (см. рис. 8.10). При необходимости изменения направления данного вектора во

время моделирования он может быть подан на вход блока Machine Environment как векторный сигнал Simulink (для этого необходимо поставить флажок в поле Input gravity as signal, см. рис. 8.10).

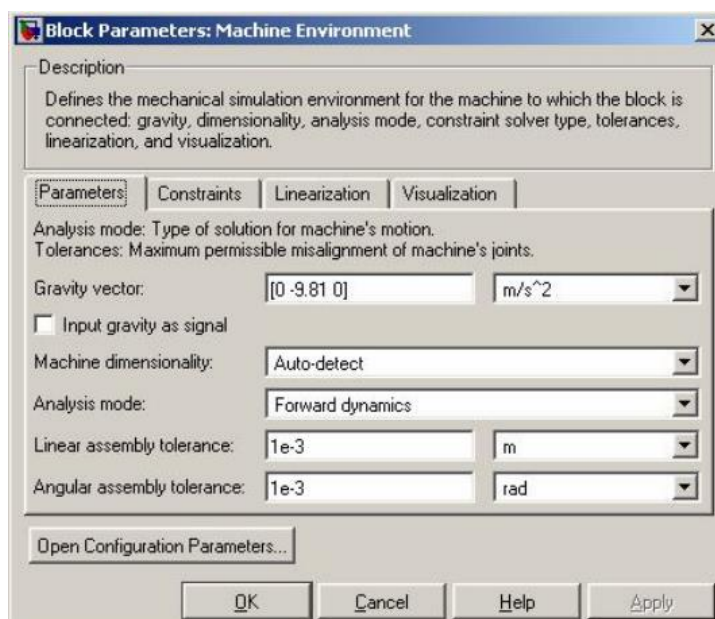


Рис. 8.10. Установка параметров механического моделирования

Предполагается, что к каждому блоку Ground, связанному с отдельным механизмом модели, должен быть присоединен свой собственный блок Machine Environment. Для того, чтобы блок Machine Environment можно было присоединить к блоку Ground, в окне настройки последнего должен стоять флажок в поле Show Machine Environment port. Все параметры блока Machine Environment, стоящие на его вкладках по умолчанию, не требуют изменения в подавляющем большинстве случаев.

2. Блоки шарнирных сочленений (Joints)

Раздел Joints (рис. 8.11) библиотеки SimMechanics – второй по значимости после раздела Bodies. Он содержит блоки шарнирных сочленений с различным числом степеней свободы, которые соединяют между собой отдельные блоки Body (а также блок Ground с блоком Body). Благодаря этому, тела (звенья механизма) получают возможность относительного движения. Раздел Joints содержит пятнадцать основных блоков, имитирующих всевозможные виды шарнирных сочленений: Prismatic; Revolute; In-plane;

Universal; Gimbal; Spherical; Planar; Cylindrical; Bearing; Telescoping; Bushing; Six-DoF; Screw; Weld; Custom Joint. Пиктограммы на блоках раздела Joints дают наглядную информацию об их назначении даже без описания.

Классификация низших кинематических пар и блоки библиотеки SimMechanics

Вид кинематической пары	Число степеней свободы	Класс пары S	Обозначение на кинематических схемах Пример использования	Блоки библиотеки SimMechanics раздел Joints
1	2	3	4	5
	3	3		 Planar Spherical Gimbal
	2	4		 In-plane Universal
	1	5	<p>Вращательная КП</p> <p>Поступательная КП</p> 	 Prismatic Screw Revolute
	0	-	<p>Неподвижное соединение звеньев</p> <p>Неподвижное звено, стойка, сварка</p>	 Weld

Кроме того, в разделе Joints присутствуют два дополнительных подраздела: Disassembled Joints (разобранные сочленения) и Massless Connectors (безинерционные соединители), которые будут рассмотрены ниже. Любой блок раздела Joints при перетаскивании его мышью в окно модели будет иметь два обязательных порта входа/выхода, которые обозначаются индексами В (Base, база) и F (Follower, последующий, см. рис. 8.9). Эти порты нельзя убрать, и они обязательно должны быть соединены: порт В с блоком Body, представляющим первое тело, порт F с блоком Body, представляющим последующее тело, которое движется относительно первого.

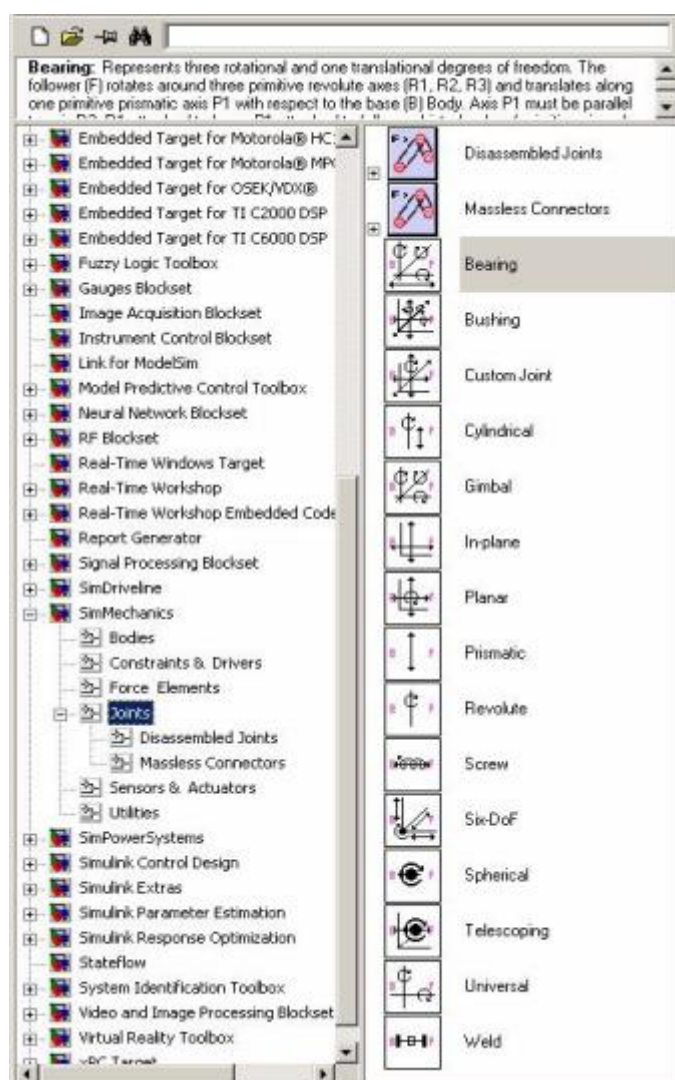


Рис. 8.11. Блоки шарнирных сочленений Joints

1. Блок **Prismatic** обеспечивает одну поступательную степень свободы тела Follower (последователь) относительно тела Base (основание). Ось, вдоль

которой может при этом двигаться телопоследователь, должна быть указана на вкладке Axes окна настройки блока (рис. 8.12) в виде вектора [X Y Z].

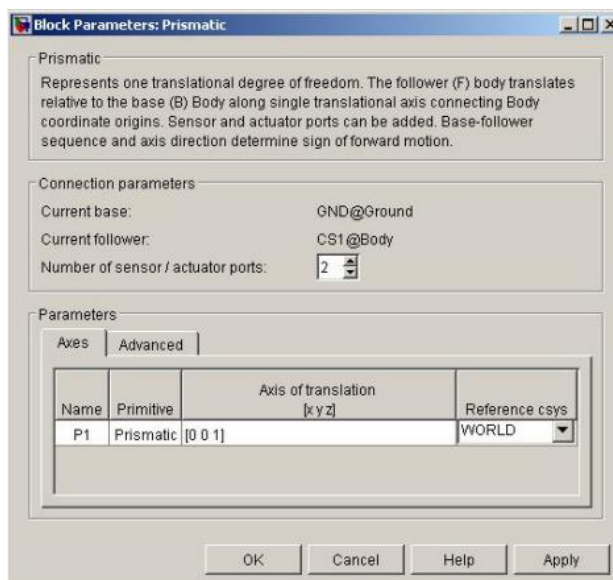


Рис. 8.12. Настройка параметров блока Prismatic

Вектор оси прямолинейного движения может быть задан в инерциальной системе координат WORLD, в системе координат, связанной с первым телом Base (основной рекомендуемый вариант), либо в системе координат, связанной с последующим телом Follower. Выбор осуществляется из выпадающего списка Reference csys. Обратите внимание, что векторы [0 0 1] и [0 0 92.47] обозначают одну и ту же ось Z.

Для того, чтобы увеличить либо уменьшить количество дополнительных портов блока, служит поле Number of sensor/actuator ports. При числе дополнительных портов, равном нулю, все равно остаются два обязательных порта B и F, о чем было сказано выше. Дополнительные порты могут использоваться для подключения к шарниру виртуальных датчиков (Sensors) и возбудителей движения (Actuators) из раздела Sensors & Actuators.

2. Блок **Revolute** обеспечивает одну вращательную степень свободы тела, следующего за блоком Revolute вокруг оси, заданной на вкладке Axes окна настройки блока (рис. 8.13) в виде вектора [X Y Z], аналогично блоку Prismatic.

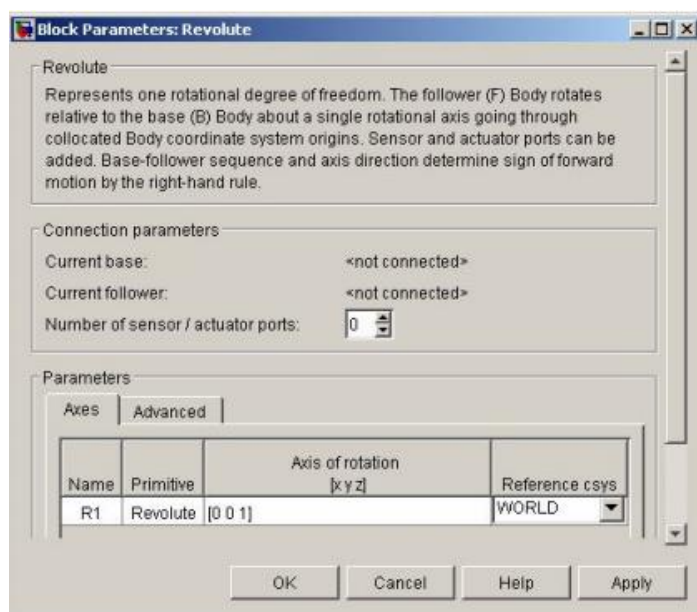


Рис. 8.13. Настройка параметров Revolute

Вектор оси вращения также может быть задан в инерциальной системе координат WORLD, в системе координат, связанной с первым телом Base (рекомендуется), либо в системе координат, связанной с последующим телом Follower.

В отличие от блока Prismatic, у которого ось прямолинейного движения и связанная с ней степень свободы названа именем P1, в окне настройки параметров блока Revolute ось вращения автоматически названа именем R1, и это не случайно. В SimMechanics все поступательные степени свободы любых блоков раздела Joints обозначаются символом P с порядковым номером после него, а все вращательные степени свободы – аналогично символом R.

3. Блоки связи механических блоков SimMechanics с обычными S-блоками Simulink (виртуальные регистраторы и возбудители движения Sensors & Actuators)

Для того, чтобы можно было измерить относительные движения тел при моделировании машин и механизмов, а также сообщить звеньям механизмов относительные движения, недостаточно блоков тел из раздела Bodies и блоков шарниров из раздела Joints. Эти блоки описывают только структуру механизма, но не сообщают ему никаких движений. Следовательно,

необходимо подключение к данным блокам других, моделирующих какие-либо воздействия и снимающих показания. Однако подключение механических блоков SimMechanics из разделов Bodies и Joints к обычным S-блокам Simulink (таким, например, как источники сигналов Sources или регистраторы Sinks) возможно только через специальные связующие блоки механических виртуальных регистраторов **тел и шарниров** (Sensors) и возбудителей тел и шарниров (Actuators).

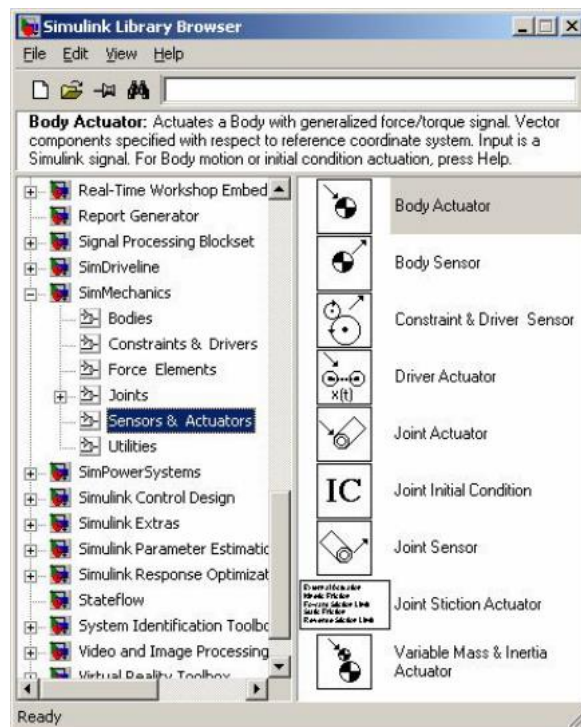


Рис. 8.14. Блоки виртуальных регистраторов и возбудителей движения Sensors & Actuators

Причины для этого следующие. Входы (и они же одновременно выходы) механических блоков тел и шарниров служат для *двухсторонней* передачи силовых взаимодействий между звеньями механизма, которые имеют место при пространственных движениях твердых тел согласно законам классической механики. Однако обычные S-блоки Simulink являются либо источниками, либо получателями, либо преобразователями *однонаправленных* сигналов, и поэтому не могут быть непосредственно подсоединены к входам/выходам механических блоков. Не случайно в блок-схемах SimMechanics на линиях соединения механических блоков нет изображений стрелок, указывающих направление воздействия, а графические изображения «входов/выходов» механических блоков имеют вид не стрелок, а квадратов с диагоналями.

Как уже было отмечено ранее, блоки типа Body и Joint, а также описанные ниже блоки из раздела Constraints & Drivers могут быть снабжены дополнительными портами для подсоединения к ним блоков Actuator и Sensor. Благодаря этому, становится возможным использовать все остальные библиотеки Simulink для формирования сигналов и перевода получаемой информации обратно в рабочее пространство Simulink. Все девять блоков раздела Sensors & Actuators (рис. 8.14) можно разделить на три группы по типу блоков, к которым они подключаются.

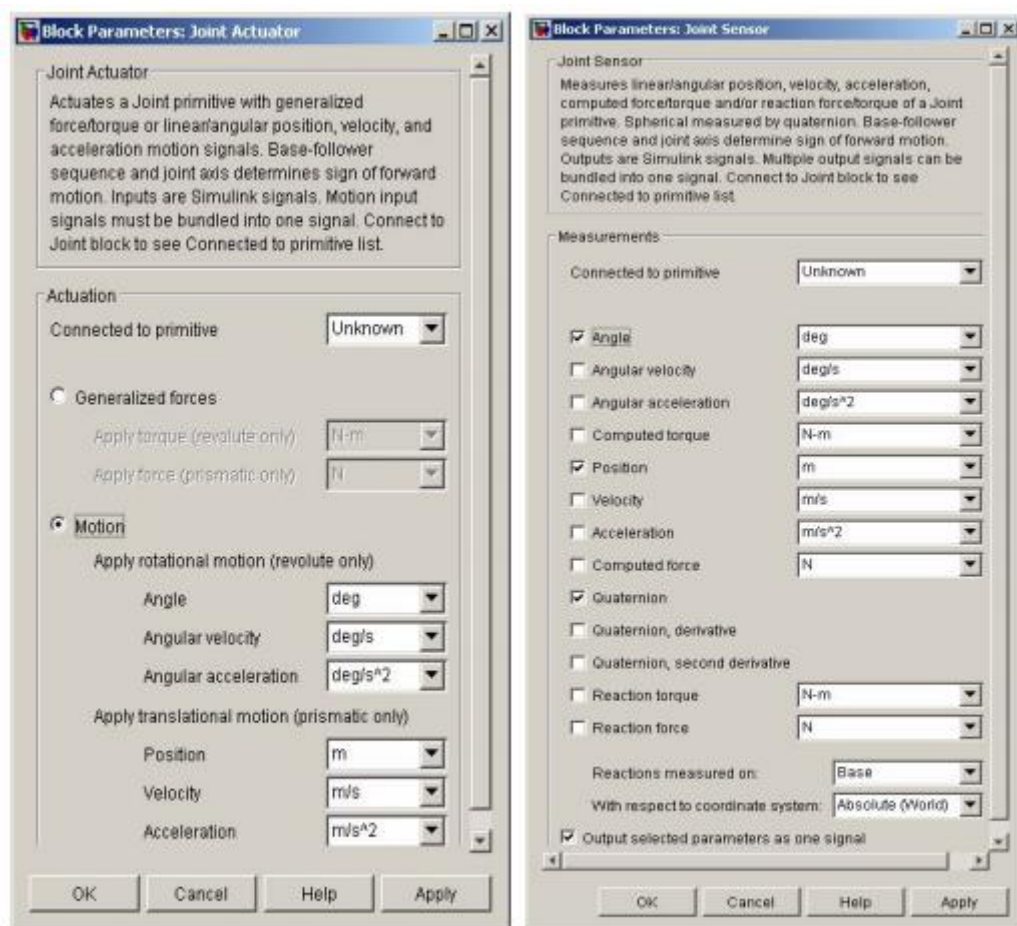
Группа 1. Блоки измерителей и возбудителей движения тел: Body Actuator; Body Sensor; Variable Mass & Inertia Actuator.

Группа 2. Блоки измерителей и возбудителей движения шарнирных сочленений: Joint Actuator; Joint Sensor; Joint Initial Condition; Joint Stiction Actuator.

Блок **Joint Actuator** позволяет задать как функцию времени:

- либо силовое воздействие между элементами примитива, имя которого выбирается из выпадающего списка в верхнем поле Connected to primitive;
- либо относительное движение элементов этого примитива.

Установка вида возбуждения (силовое воздействие Generalized forces или движение Motion) осуществляется при помощи переключателя (рис. 8.15). Выбрать примитив можно лишь в том случае, если блок Joint Actuator уже присоединен к дополнительному порту какого-либо блока Joint. Когда выбран вид возбуждения Motion, относительное движение частей примитива должно быть подано в виде векторного сигнала Simulink вида [перемещение; скорость; ускорение] на вход блока Joint Actuator. В случае силового возбуждения на вход подается скалярный сигнал.



Настройка параметров блока
Joint Actuator

Настройка параметров блока
Joint Sensor

Рис. 8.15. Настройка блоков

Блок **Joint Sensor** позволяет в общем случае измерить следующие характеристики относительного движения частей примитива, выбранного из выпадающего списка в верхнем поле *Connected to primitive* (рис. 8.15):

- угол (*Angle*) – угол поворота части примитива, соединенной с телом *Follower*, относительно его части, соединенной с телом *Base*;
- относительную угловую скорость (*Angular velocity*);
- относительное угловое ускорение (*Angular acceleration*);
- вычисленный момент (*Computed torque*) – полный момент сил, вызывающий относительное угловое ускорение;
- позицию (*Position*) – перемещение части примитива, соединенной с телом *Follower*, относительно его части, соединенной с телом *Base*;
- скорость (*Velocity*) – относительная скорость;
- ускорение (*Acceleration*) – относительное ускорение;

- вычисленную силу (Computed force) – полную силу, вызывающую относительное ускорение;

- кватернион (Quaternion) – вектор из четырех компонент, описывающих текущее угловое положение частей сферического примитива.

Кватернион q определяется по формуле

$$q=[nX \cdot \sin(\theta/2); nY \cdot \sin(\theta/2); nZ \cdot \sin(\theta/2); \cos(\theta/2)],$$

где nX , nY , nZ – компоненты единичного вектора, определяющего ось поворота тела Follower на сферическом примитиве; θ – угол поворота вокруг указанной оси;

- производная кватерниона (Quaternion, derivative) – вектор из четырех компонент, каждая из которых является производной по времени от соответствующей компоненты вектора Quaternion;

- вторая производная кватерниона (Quaternion, second derivative)

- вектор из четырех компонент, каждая из которых является второй производной по времени от соответствующей компоненты вектора Quaternion;

- момент реакции (Reaction torque) – момент реакции относительно оси примитива;

- реакция (Reaction force) – сила реакции вдоль оси примитива. Для выбора необходимых измеряемых параметров служит простановка флажков в соответствующих полях. Поскольку примитив, выбранный в поле Connected to primitive, может быть одного из трех типов (поступательного P, вращательного R или сферического S), не все перечисленные характеристики могут быть измерены для конкретного примитива. Некоторые поля в окне настройки при выборе примитива становятся неактивными.

Блок **Joint Initial Condition** занимает особое положение среди других блоков раздела Sensors & Actuators. Он задает начальное относительное положение и начальную относительную скорость двух частей элементарных примитивов R-типа (Revolute) или P-типа (Prismatic) какого-либо шарнира. При помощи одного блока Joint Initial Condition могут быть заданы начальные условия положения и скорости для всех примитивов шарнира, к которому он

подключен, либо только для части примитивов шарнира. Выбор примитивов осуществляется простановкой флажков в полях колонки Enable окна настройки блока (рис. 8.16). То есть начальные условия задаются по отдельным степеням свободы шарнира. Начальное положение – в полях колонки Position, начальная скорость – в полях колонки Velocity.

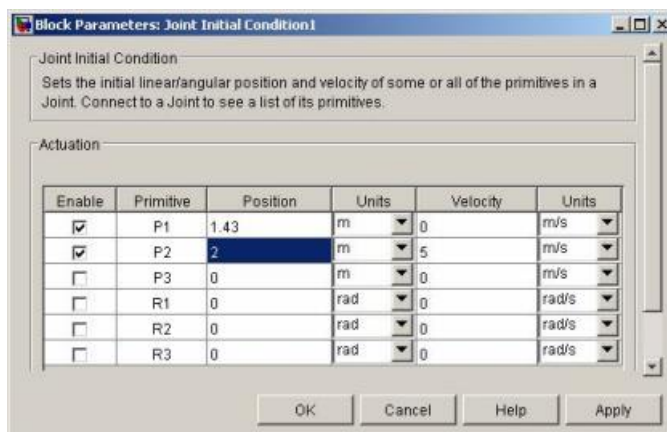


Рис. 8.16. Настройка параметров блока Joint Initial Condition

Основу блок-схемы любого механизма в SimMechanics составляет цепь звеньев типа Ground-Joint-Body-Joint-Body-Joint-Body. Как вариант, вместо шарниров типа Joint в цепи могут присутствовать блоки связей Constraints & Drivers. Эти цепи могут быть с открытой или с закрытой топологией, то есть замкнутые или разомкнутые.

Чтобы избежать ошибок при моделировании, необходимо быть уверенным, что топология построенной блочной диаграммы SimMechanics правильная. Блочная диаграмма будет правильна с топологической точки зрения, если все механизмы, из которых она состоит, адекватны. Механизм же адекватен, если адекватно его основное дерево (цепь звеньев, указанная выше). Таким образом, чтобы определить, является ли модель адекватной, сначала необходимо по- 48 лучить основное дерево каждого механизма, составляющего модель, и затем определить адекватность каждого дерева. Для получения основного дерева механизма необходимо удалить (условно) из модели механизма все блоки, кроме блоков тел и шарниров Ground, Joint и Body и открыть каждый замкнутый контур в механизме при их наличии.

Чтобы открыть замкнутый контур, необходимо удалить из этого контура блок шарнира Joint или связи (при наличии блоков из раздела Constraints & Drivers).

Топологическая адекватность основного дерева гарантируется следующими условиями: основное дерево должно иметь по меньшей мере один блок основания Ground, связывающий механизм с инерциальной системой координат; каждый блок шарнира должен быть обязательно связан с двумя блоками тел; каждый блок тела не основания Body должен иметь собственный путь к блоку основания Ground; каждый блок тела не основания Body в конце цепи звеньев тел должен иметь массу или инерционный момент, отличные от нуля (каждая поступательная степень свободы должна нести массу, отличную от нуля, и каждая вращательная степень подвижности – инерционный момент, отличный от нуля. Это предотвращает непрерывное ускорение, когда применяются силы и крутящие моменты).

Кроме перечисленных выше существует еще одно условие работоспособности механизма в модели SimMechanics. Механизм должен иметь по меньшей мере одну степень подвижности. Свободное физическое тело имеет шесть степеней свободы: три поступательных и три вращательных. Но в механизме при добавлении связей между телами посредством шарнирных сочленений и блоков Constraints & Drivers независимые степени свободы снижаются до меньшего количества. Также снижается количество степеней свободы, если движение механизма ограничивается до двух пространственных измерений в настройках блока Machine Environment.

Блок любого тела Body сам по себе не имеет степеней свободы. При подключении к нему блоков шарнирных сочленений Joint это добавляет степени свободы для всего механизма. Подключение блоков связей Constraints & Drivers к телам или шарнирам, напротив, снижает степени свободы механизма.

Число степеней свободы плоского механизма

$$W = H \cdot n - \sum_{i=2}^{H-1} (H - i) \cdot p_i,$$

где H - число степеней подвижности свободного твердого тела; n - число подвижных звеньев в механизме $n = k - l$; k - общее число звеньев в механизме; i - число подвижностей в кинематической паре; p_i - число кинематических пар с i подвижностями.

Удобно пользоваться выражением, которое является следствием первой формулы и предназначено для вычисления числа степеней свободы плоского механизма:

$$W_n = 3(k - 1) - 2P_5 - P_4,$$

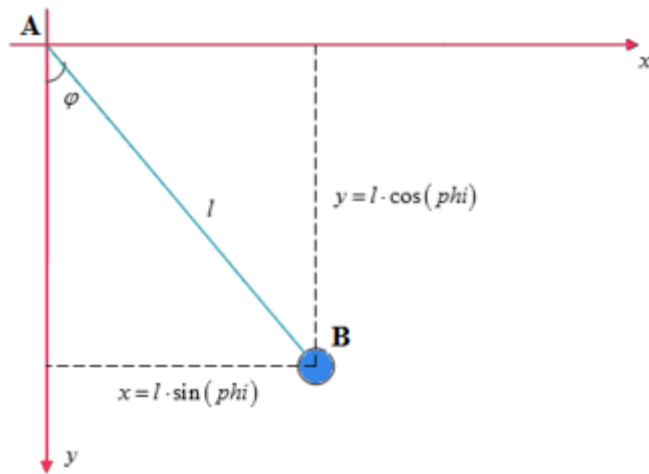
где k - общее число звеньев в механизме; P_5 - количество пар пятого класса; P_4 - количество пар четвертого класса.

Для определения числа ограничений движения C необходимо просуммировать количество ограничений по каждому блоку типа Constraints или Drivers механизма. Для этого нужно воспользоваться табл. 2, в которой приведены числа ограничений, создаваемых каждым блоком из раздела Constraints & Drivers.

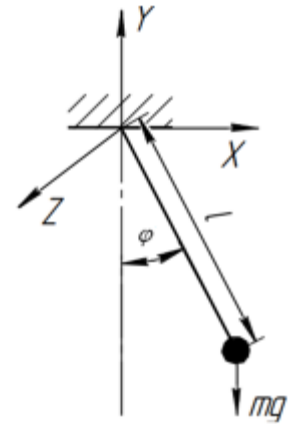
Количество ограничений движения, создаваемых блоками связей Constraints & Drivers

Блоки Constraints		Блоки Drivers	
Gear	1	Angle	1
Parallel	2	Distance	1
Point-Curve	2	Linear	1
		Velocity	1

Пример 8.1. Смоделировать работу математического маятника, подвешенное на нерастяжимой нити длиной l и отклонено на некоторый угол φ от положения вертикали.



а)



б)

Рис. 8.16.

Решение:

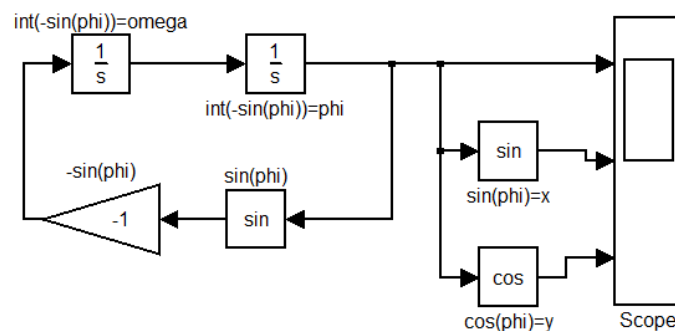
1. С помощью моделирования в Simulink.

Движение математического маятника описывается $\frac{d^2\varphi}{dt^2} + \sin(\varphi) = 0$, где

φ - угол отклонения маятника от положения равновесия $x=0$, т.е. точки А (рис. 8.16(а)). Перепишем это уравнение относительно угла φ :

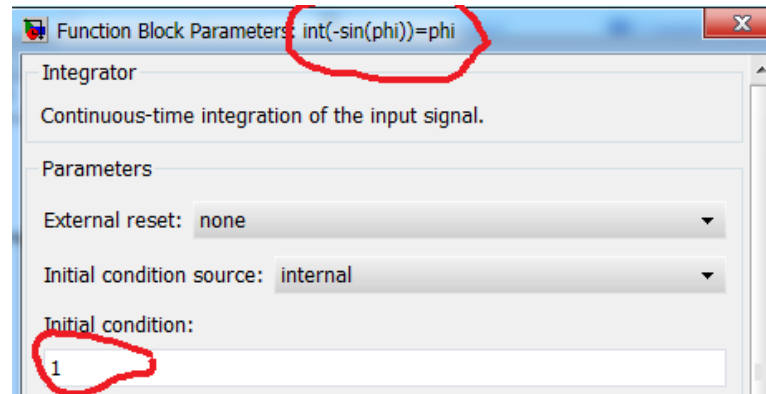
$$\frac{d^2\varphi}{dt^2} = -\sin(\varphi); \quad \varphi = \int \left(\int -\sin \varphi dt \right) dt.$$

В данном уравнении видим зависимость изменения угла отклонения маятника с течением времени.

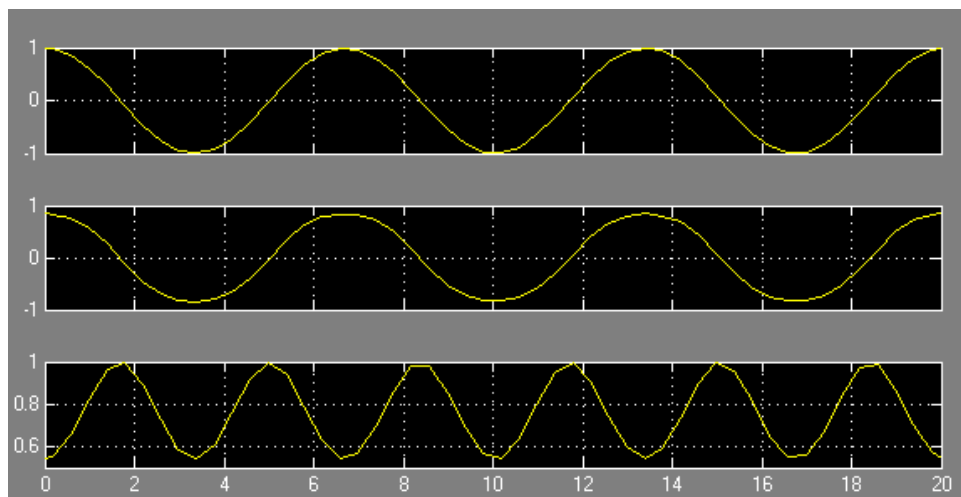


По умолчанию на интеграторы подаются начальные условия $\omega=0$ (угловая скорость) и $\varphi=0$. В таком случае колебание маятника не будет. Нужно задать начальную угловую скорость или начальный угол отклонения.

Изменим начальный угол отклонения маятника:



Результатом симулирования будут графики: изменение угла отклонения φ ; горизонтальная составляющая движения x и вертикальная составляющая движения y .



2. С помощью моделирования в SimMechanics

Математический маятник является системой с одной степенью свободы. Его кинематический анализ показывает, что невесомый стержень маятника длиной l (рис. 8.16) соединен со стойкой А через вращательную кинематическую пару пятого класса, поэтому для наложения кинематических ограничений рационально воспользоваться блоком Revolute библиотеки SimMechanics (SM), который обеспечивает одну вращательную степень подвижности относительно выбранной оси. В данном случае было принято, что маятник совершает колебания относительно оси Z. Следующим шагом, который необходимо сделать при создании любой модели – это указать

«землю», т.е. точку отсчета (стойку), в которой будет располагаться глобальная система координат модели. Для указания «земли» необходимо использовать блок Ground библиотеки SM. В механизмах количество стоек различно. В модели может быть использовано столько стоек, сколько требует структурный анализ. Однако, при моделировании в SM, необходимо указать начало отсчета для всей системы и подключить к нему вектор гравитационных сил, который задаётся блоком Machine Environment. В блоке Machine Environment гравитационные силы задаются относительно выбранной оси. При моделировании маятника начало координат помещено в точку А и совмещено со стойкой, а вектор гравитационных сил направлен в отрицательном направлении вдоль оси Y .

Расчеты при начальных условиях: $\varphi = 20^\circ$; $AB = 10 \text{ мм}$; $A_x = 0$; $A_y = 30$.

$$B_x = A_x + AB \cdot \sin(\varphi) = 3,42 \text{ мм}; \quad B_y = A_y - AB \cdot \cos(\varphi) = 20,603 \text{ мм};$$

Центр тяжести (без груза):

$$AB_x = A_x + \frac{AB}{2} \cdot \sin(\varphi) = 1,71 \text{ мм}; \quad AB_y = A_y - \frac{AB}{2} \cdot \cos(\varphi) = 25,301 \text{ мм}.$$

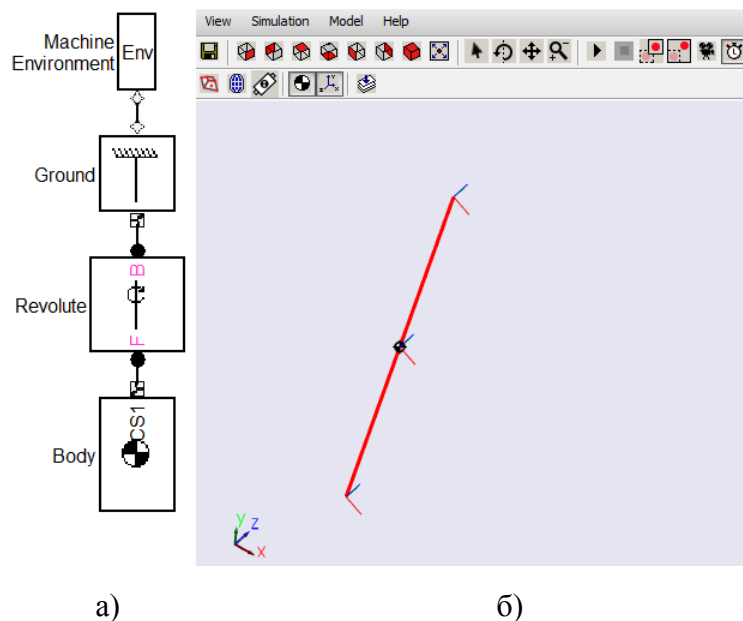
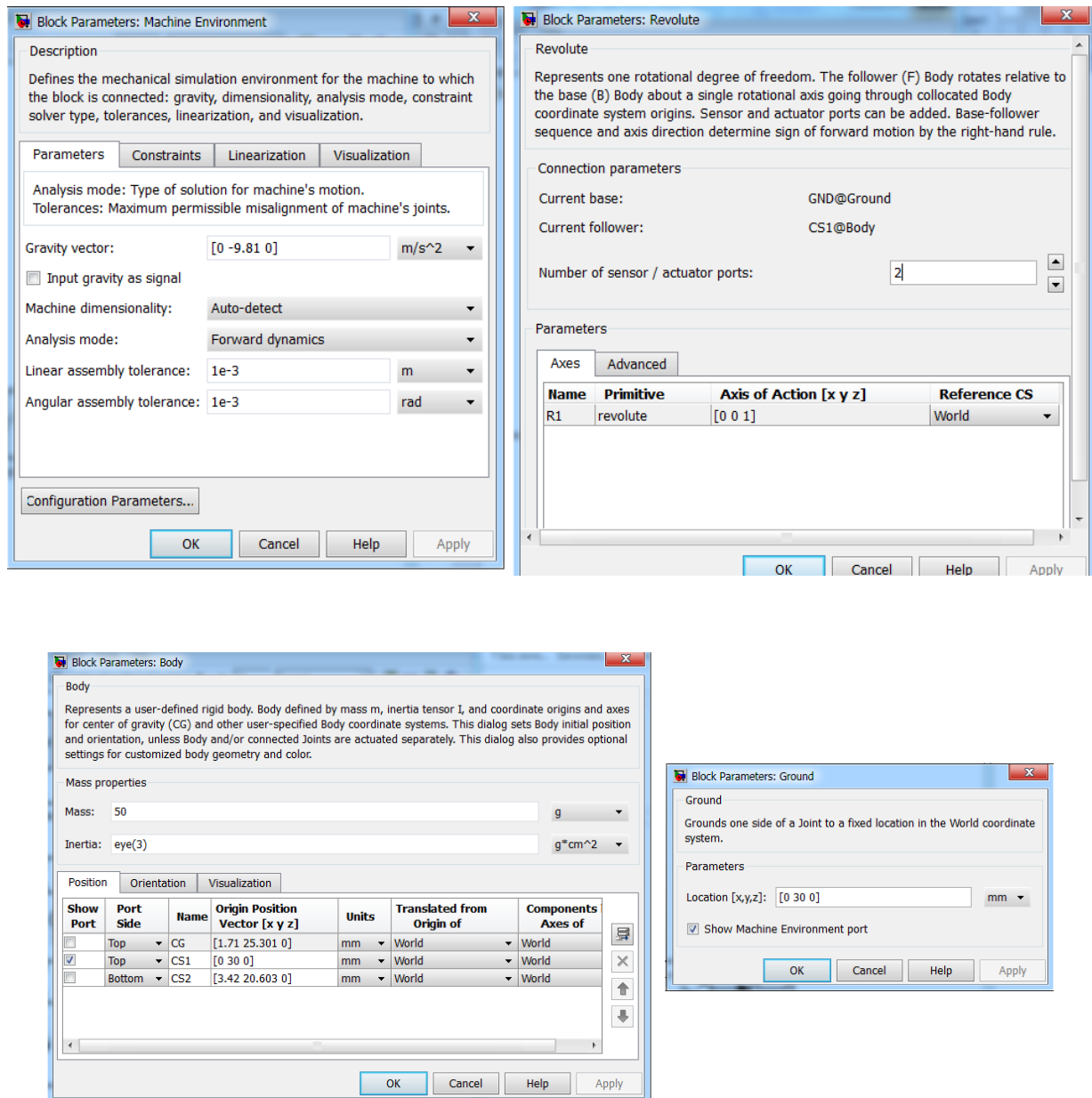


Рис. 8.17. Структурная модель SimMechanics маятника (а); анимационная модель SimMechanics маятника (б)

Параметры настройки блоков согласно расчетам:



Для получения анимации необходимо в меню Simulink вызывать окно настройки параметров модели: пункт Simulation/Configuration Parameters... далее в левой части окна в дереве выбрать Simscape\SimMechanics и установить Visualization\Show animation during simulation. После проделанных действий при запуске моделирования автоматически откроется окно Machine for model: с анимацией положения всех тел модели. Чтобы представить подвижные тела модели в виде сфер (эллипсоидов -- в зависимости от их тензора инерции) необходимо выбрать в меню окна анимации пункт Model/Body Geometries/Ellipsoids. Для замедления анимации можно вызвать

окно параметров через меню Simulation/Control Animation Speed и установить там задержку Delay per frame (ms).

Обратите внимание, что картинку анимации можно приближать или отдалять мышкой при выборе пункта меню View/Zoom или можно воспользоваться View/Fit to view.

После моделирования получаем следующие результаты (рис. 8.18) (желтый график – положение, красный - скорость), которые согласуются с физическим смыслом данной системы.

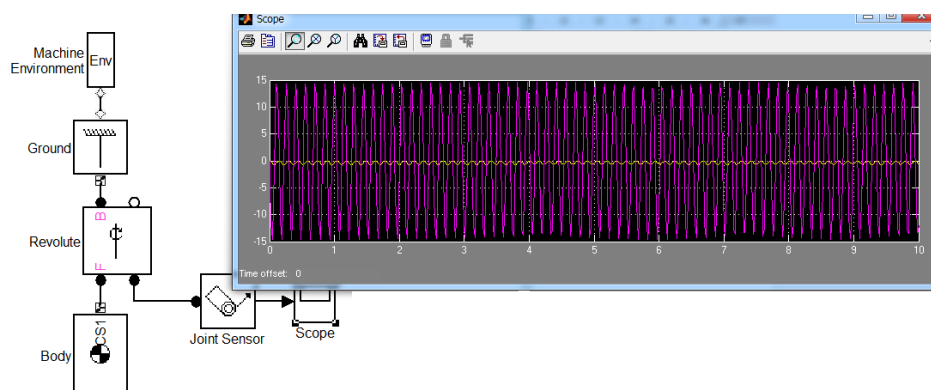


Рис. 8.18. Результат работы построенной модели